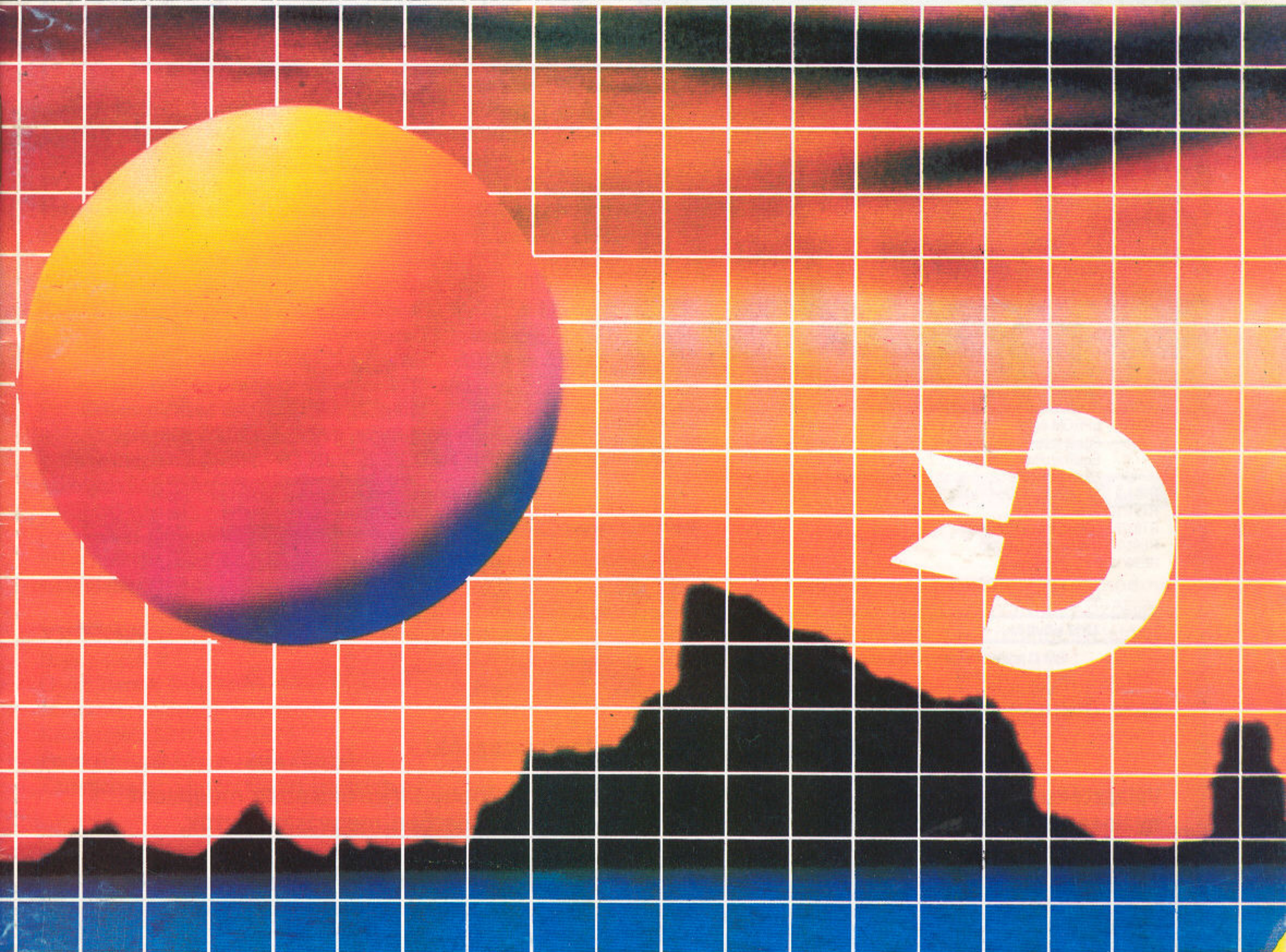
Az Országos Commodore Egyesület lapja

C újság

VIII. évfolyam **1993/7–8**



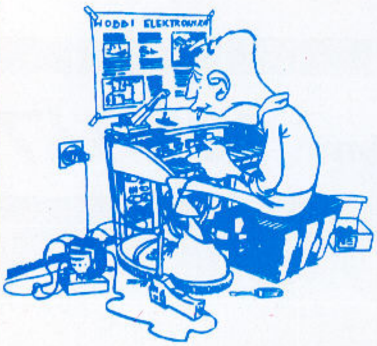
Directory kezelő

80 karakteres jegyzettömb

Új sorozat

AMIGA ROVAT

Szuper ajánlat!



HOBBI ELEKTRONIKA

Urbán István mérnök áramköreinek szaküzlete

Budapest VII., Dózsa György út 16. (Dózsa–Jobbágy sarok)
Nyitva: H–P 10–17-ig Tel./fax: 122-8892
(Zárás után üzenetrögzítő)

Több, mint tíz éve írok a Rádiótechnikában, vezetem a „Zenei hobby elektronika" fejlécű rovatot. A közölt témák nagyrészéhez szaküzletemben - működő minta alapján - vásárolhat panelokat, részegységeket, egységcsomagokat.

KÍNÁLATUNKBÓL:

SZÁMÍTÁSTECHNIKA:

	C64 bővítők:	egys.	éleszt.
RT89/3	PLOFI Datassette cartridge	1200 Ft	1500 Ft
	PLOFI Datassette/promon	1300 Ft	1700 Ft
	PLOFI Datassette/help	1300 Ft	1700 Ft
RT89/10	PLOFI Fastload cartridge	1300 Ft	1700 Ft
	PLOFI Fastload speedtape	1400 Ft	1900 Ft
RT90/3	PLOFI Simon's cartridge	1300 Ft	1700 Ft
	FINAL III cartridge		3450 Ft
	ACTION Replay VII		3450 Ft
RT89/11	Fényceruza, szoftverrel	950 Ft	1250 Ft
RT90/8	Hangdigitalizáló + szoftver	1350 Ft	1650 Ft
HE90/8	Hangkapcsoló	300 Ft	
RT89/4	EPROM-égető	3400 Ft	4500 Ft
	Égető szoftver lemezen	1300 Ft	
	Égető szoftvercartridge	1300 Ft	1700 Ft
	User csatlakozó	300 Ft	
RT91/3	EPROM-bank (256 Kbájt)	2900 Ft	4000 Ft
RT91/10	IC teszter	3000 Ft	4500 Ft
RT91/7	Datassette gyorsmásoló	600 Ft	
HE91/7	Datassette fejbeállító	350 Ft	
HE90/12	CPU stop + reset	400 Ft	
....	PAGEFOX szövegszerkesztő		3900 Ft
....	Mini EPROM-bank	2300 Ft	3000 Ft

	IBM bővítők:	egys.	éleszt.
RT91/4,5	IBM IC teszter + szoftver	5900 Ft	9800 Ft
RT91/11,12	IBM EPROM-égető + szoftver	4800 Ft	9900 Ft
RT91/6	48 csat. I/O kártya	3500 Ft	

HE92/2	User-Centronics csatlakozó	950 Ft
RT92/3	TTL IC katalógus lemezen	600 Ft
	CMOS IC katalógus lemezen	600 Ft
	Dióda katalógus lemezen	600 Ft
	Tranzisztor katalógus lemezen	600 Ft

ZENE – HANGTECHNIKA:

RT87/10	Fuzz-box torzító	670 Ft
RT90/1	KORG DST-1 torzító	1200 Ft
....	KORG DST-3 torzító	1400 Ft
HE90/1	KORG OVERDRIVE torzító	920 Ft
RT87/12	Vau-vau gitáreffekt	640 Ft
RT88/9	Sztereó tremoló	980 Ft
RT88/7	Shifter	1200 Ft
HE90/7	Kiszajú gitárelőerősítő	700 Ft
RT89/5	Sztereó előerősítő	1500 Ft
HE91/12	Hangfrekvenciás erősítő	490 Ft
RT89/8	2 × 14 W sztereó erősítő	990 Ft
RT89/12	2 × 40 W sztereó erősítő	1800 Ft
RT89/1	100 W-os erősítő	1600 Ft
RT90/3	Ritmusgép 32 ritmussal	2500 Ft
RT90/10	Dobszintetizátor	3500 Ft
RT88/3	Süvöltő gitáreffekt	780 Ft
RT87/11	Ringmodulátor	995 Ft
RT92/4	Oktávemelő	700 Ft
....	Visszhangosító MN3005-tel	4800 Ft

VIDEÓS TÉMÁK:

RT92/6	RGB generátor	1760 Ft
RT92/11	PAL kóder	1400 Ft

FÉNYTECHNIKA:

RT88/10	Diszkófény II (triak nélkül)	1400 Ft
	Diszkófény II (triakkal)	3000 Ft
RT88/12	Fényorgona	2000 Ft
RT90/2	Programozható futófény	1200 Ft
RT91/1	8 × 8-as fénymátrix + 16 K EPROM	2000 Ft
	8 × 8-hoz LED panel 10 mm-es LED-ekkel	2000 Ft
RT89/6	Kivezérlésjelző	700 Ft
RT91/8	Sziporkázó színgyűrű	650 Ft
	Kétszínű LED	48 Ft
HE91/10	Karácsonyi fényjáték	700 Ft
HE92/8	Knight Rider futófény	950 Ft

EGYÉB HOBBI TÉMÁK:

RT88/10	Dallamgenerátor Z80-nal	1500 Ft
HE91/1	Dallamgenerátor UM3481-84	800 Ft
HE91/5	Dallamgenerátor UM66Txx	480 Ft
HE91/4	Beszédfelismerő VCP200-zal	750 Ft
	VCP200	1950 Ft
HE91/2	Kutyaugatás szintetizátor	800 Ft
RT91/8	Eb-vezérlő	400 Ft
HE91/6	Szúnyogriasztó	700 Ft
HE91/9	Multisziréna	480 Ft

Az árak a 25% ÁFA-t tartalmazzák!

Választékunk folyamatosan bővül! A Rádiótechnikában, a Hobby Elektronikában megjelent nyák-tervek alapján készült nyomtatott áramköri lapok az üzletben megvásárolhatók.

MŰSZERVÁSÁR
Bontott anyagok, használt műszerek böngészdéje



Állandóan változó készlet!

Vidéki olvasóknak segít a szerző levelező, egységcsomagküldő szolgáltatása: a megrendelt csomagot postán utánvéttel elküldöm. Telefonon/faxon és levélben is rendelhet.
A HOBBI ELEKTRONIKÁ-hoz nem kell hosszú levél. Rendelését néhány sorban, egyértelműen közölje.

Levélcím: 1656 Budapest Pf. 50.

Az üzletben megvásárolhatók az RT évkönyvei, a Rádiótechnika és a Hobby Elektronika korábbi számai.

SZERENCSEJÁTÉK RT

MIT, HOGYAN, HOL, MIKOR?

EGYESÜLETI ÜGYEK: Egyesületünknek tagja lehet mindenki, aki a tagsági díjat befizeti. A tagdíjat személyesen az egyesület irodájában (1025 Budapest, Vöröstorony utca 29. Telefon: 1-76-22-57), vagy átutalással az MNB 217-98 292, OTP 565-3610-8 számlára lehet befizetni. Megrendelés esetén számlát küldünk.

Pötyögőszolgálatunk valamint a szervizkedvezmény és az apróhirdetés lehetősége tagjaink rendelkezésére áll.

A DEÁKPÁHOLY tagjai minden hónapban megkapják a C-újságot, a tagsági díj egy egy évre 890 Ft.

A PLUSZPÁHOLY tagjai minden hónapban megkapják a C-újságot, és kapnak havonta 3 db vásárlási utalványt. A tagsági díj egy évre 2050 Ft.

A SZUPERPÁHOLY tagjai havonta 15 példányt kapnak a C-újságból, és ezzel havonta 15x3 db vásárlási utalványt is. Az éves tagsági díj 24 000 Ft.

ÜGYFÉLFOGADÁS: Minden kedden és csütörtökön 12—16 óra között várjuk tagjainkat és az érdeklődőket.

PÖTYÖGŐSZOLGÁLAT: Az újságban megjelenő programokat másolja a megrendelők részére. Megrendelhető személyesen az egyesület irodájában vagy postai utánvéttel. Postacím: 1388 Budapest 62., Postafiók: 86.

APRÓHIRDETÉS: Az egyesületi tagoknak ingyen áll rendelkezésére. Nem tagoknak a hirdetés ára 100 forint. A hirdetés módja: az újságban megjelenő nyomtatvány kitöltésével.

A C-ÚJSÁG RÉGEBBI SZÁMAI megvásárolhatók az egyesület irodájában, vagy megrendelhetők utánvéttel.

Kedvezményes ár! Tagoknak olcsóbb!

Az újságban eddig megjelent programok gépenként összegyűjtve megrendelhetők. VC 20, C16, PLUS/4, C128, C64. További felvilágosítást is adunk a 1-76-22-57-es telefonszámon vagy levélben!

Vidéki pluszpáholy-tagjaink háromhavi tikett összegyűjtésekor igénybe vehetik a NOVOTRADE 2C Áruház csomagküldő szolgálatát.

VIDÉKEN TOVÁBBI INFORMÁCIÓK KAPHATÓK:
Baja, AXIS Kft.,
Győri Bartók Béla Művelődési Ház,
Jászberényi Városi Könyvtár,
Kecskemét, SZIGMA—BIT,
Pécsi Apáczai Csere János Gimnázium,
Zalaegerszegi Ságvári Endre Gimnázium.

Az Országos Commodore Egyesület módszertani kiadványa

Egyesületi iroda és szerkesztőség:
1025 Budapest, Vöröstorony utca 29. Telefon: 1-76-22-57
Felelős kiadó: Horváth Judit, az egyesület elnöke
Főszerkesztő: Rados Péter, az OCE főtitkára
Felelős szerkesztő: dr. Horváth András
Művészeti szerkesztő: Bausz Sándor
Levélcím: Commodore Újság, 1388 Budapest, 62. Pf.: 86.
Index: ISSN 0237-756 X
Terjeszti a Magyar Posta
Megvásárolható a hírlapárusoknál
93.0161 MSZH Nyomda és Kiadó Kft., Budapest
Felelős vezető: Nagy László

Az Országos Commodore Egyesület pályázatot hirdet. A pályázaton bárki részt vehet. A pályázat tárgya: C64-en vagy C+4-en futó, saját készítésű felhasználói program. A program a gép saját BASIC nyelvén készüljön, terjedelme ne haladja meg a 300 sort. A programot kérjük lehetőleg lemezen (esetleg kazettán) beküldeni. A pályázatnak tartalmaznia kell a program felhasználói útmutatóját, valamint a program belső működését ismertető leírást.

A pályázat helyezettjei számítógépeket és tartozékokat nyernek.

A pályázat beküldési határideje: 1993. szeptember 20.

A pályaműveket kérjük az egyesület postacímére: 1388 Budapest 62, Postafiók 86. eljuttatni. A borítékra írják rá: „Alkalmazói program pályázat"

Pályázatunknak nem titkolt célja, hogy egyesület tagjai és a lap olvasói olyan programokkal lepjék meg egymást, melyeket az iskolában, otthon, munkahelyen, kisvállalkozásban jól tudnak használni. A programok használhatóságát nagymértékben meghatározza a dokumentáció minősége. Ezért felhívjuk pályázóink figyelmét az 1993/4-es számunk „Még többet ésszel" rovatában leírtakra.

Országos Commodore Egyesület

C-64 C+4 AMIGA PC

tulajdonosok!

Várunk benneteket az Országos Commodore Egyesület klubdélelőttjén a Havanna Közösségi Házban minden hónap 3. vasárnapján, délelőtt 9 órától. Gépet és hosszabítót hozzatok!
Cím: 1181 Budapest, Kondor Béla sétány 8.
Megközelíthető: a metró kőbánya–kispesti végállomásától a piros 136-os busszal.
Legközelebb:

július 18-án,
augusztus 15-én.

FELHÍVÁS

Tisztelt Tagtársunk, Olvasóink!
Kérjük, hogy ha rendelkeznek olyan működő számítógéppel, vagy jó állapotban lévő bármilyen számítógéptartozékkal, amire nincs szükségük, akkor hozzák el egyesületi irodánkba (minden héten kedden és csütörtökön 11—15 óra között).
Mi összegyűjtjük és eljuttatjuk azokat határainkon túli (erdélyi, kárpátaljai és szlovákiai) magyar iskolákhoz.
Köszönettel:

Országos Commodore Egyesület

Tisztelt Szerkesztőség

12 éves vagyok, és a C/64-es gépemen programozok BASIC-ben. Nemrég lettem tagja a Commodore Egyesületnek, és nagyon sok hasznát veszem a C-újságnak.

Az 1993. évi első szám hetedik oldalán van egy program „Szorzótábla helyett" címmel. Ennek van egy hiányossága: végtelen sokáig gondolkodhatunk. Az általam küldött programban van időkorlátozás. Használata a következő: Miután elindítottuk, megkérdezi mennyi időt adjak a feladat megoldására. Itt be kell írni a választ, és megnyomni a RETURN-t. Ezután felírja az első feladatot. Erre be kell írni a választ, és megnyomni a RETURN-t. Ha nem írtunk be időben, kiírja: lejárt az idő. Ha időn belül írtuk be, válaszunk után bármelyik billentyűvel kérhetjük a következő szorzást, kivéve a V betűvel nem, mert ekkor kiírja a találati arányt, és vége a programnak.

Most röviden leírom a program szerkezetét

100—160 Megkérdezi, mennyi időt adjon. Ha ide nem írunk be semmit, csak megnyomjuk a RETURN-t, nincs korlátozva az idő.

180—230 Kitalál egy feladatot a véletlenszám generátor segítségével. Ha nincs korlátozva az idő, tovább megy, ellenkező esetben az IDŐ alprogramra ugrik.

250—300 Időkorlátozás nélkül kér két számjegyet, ha csak egyet kap, a RETURN-nal lehet továbbmenni.

310—370 A két szövegváltozóból egy számváltozót készít.

380—465 Ellenőrzi és kiértékeli válaszunkat.

470—560 Ez a program lényege, az időkorlátozás. Úgy működik, hogy a 126 sorban beadott válaszunkat összehasonlítja az elején lenullázott időváltozóval, és közben a GET utasítás a billentyűzetet kérdezi le.

600— A találati eredményt mutatja meg.

A program Commodore C64-es gépre készült, de a 70—90-es sorokat megváltoztatva más olyan gépre is jó, amin van TIS változó.

Remélem programomat érdemesnek találják a közlésre.

Tisztelettel:
Jánossy Gergely

```
0 REM AZ 1-RE VEGZODO SOROKAT NEM KELL
1 REM BEGEPELNI
2 REM
10 REM ******************
20 REM * SZORZOTABLA C64 *
30 REM *BOVITETT VALTOZAT*
40 REM * JANOSSY GERGELY *
50 REM *  A C= UJSAGNAK  *
60 REM ******************
70 POKE53280,0:POKE53281,0
80 POKE646,15
90 POKE53272,21
100 PRINTCHR$(147)
110 PRINT"QQQMENNYI IDOT ADJAK A FELADAT"
111 REM 3[CRSR LE]
120 PRINT"MEGOLDASARA? /MASODPERCBEN/
125 PRINT"HA NULLAT IRSZ BE,VEGTELEN SOK IDOD VAN."
126 INPUTX$
150 IF LEN(X$)=1 THEN ID$="00000"+X$
160 IF LEN(X$)=2 THEN ID$="0000"+X$
170 :
180 REM FELADAT
190 V1$="":V2$=""
200 R1=INT(8*RND(1)+2)
210 R2=INT(8*RND(1)+2)
220 PRINT"♥QQQ]]";R1;".";R2;"=?"
221 REM [SHIFT+CLR HOME] 3[CRSR LE] 2[CRSR JOBBRA
230 IF VAL(ID$)>0 THEN GOTO 470:REM >IDO
240 :
250 GET V1$
260 IF V1$="" THEN 250
290 GET V2$
300 IF V2$="" THEN 290
310 REM *** ELLENORZES ***
320 IF V2$=CHR$(13) THEN GOTO 360:REM >EGYJEGYU
330 V=VAL(V1$)*10+VAL(V2$)
340 WAIT198,255
350 GOTO 380:REM ->KIIRAS
360 REM EGYJEGYU
370 V=VAL(V1$)
380 REM KIIRAS
390 PRINTV;"-T IRTAL."
400 IFV=R1*R2 THEN A$="█UGYES VAGY"+"█":AJ=AJ+1
401 REM [CTRL+6] [C=+8]
410 IF V<>R1*R2 THEN A$="█CSACSI VAGY!HELYESEN:"+STR$(R1*R2)+"█":AR=AR+1
411 REM [CTRL+3] [C=+8]
```

```
420 REM EREDMENY
425 IF A$="LEJART AZ IDO!!!" THEN AR=AR+1
430 PRINT"QQQ]]]]R";A$
431 REM 3[CRSR LE] 4[CRSR JOBBRA] [CTRL+9]
440 POKE190,0:
450 GET X$:IF X$="" THEN 450
460 IF X$<>"V"THEN GOTO180: REM ->FELADAT
465 GOTO 600
470 REM *** IDO ***
480 TI$="000000"
500 IF V1$="" THEN IF V2$=""THEN GET V1$
510 IF V1$<>"" THEN GET V2$
520 IF V2$<>"" THEN TI$=ID$
530 IF TI$<>ID$ THEN 500
540 IF V1$="" THEN A$="LEJART AZ IDO!!!"+STR$(R1*R2):GOTO 420:REM ->EREDMENY
550 IF V2$="" THEN A$="LEJART AZ IDO!!!"+STR$(R1*R2):GOTO 420:REM ->EREDMENY
560 GOTO310:REM ->ELLENORZES
570 :
580 :
600 PRINT"♥QQQ]]REMELEM SOKAT TANULTAL!"
601 REM [SHIFT+CLR HOME] 3[CRSR LE] 2[CRSR JOBBRA]
610 PRINT"QQ";AR+AJ;"FELADATBOL"
611 REM 2[CRSR LE]
620 PRINT"QQ]";AR;"NEM SIKERULT ES"
621 REM 2[CRSR LE] [CRSR JOBBRA]
630 PRINT"QQ]";AJ;"SIKERULT"
631 REM 2[CRSR LE] [CRSR JOBBRA]
632 REM 3[CRSR LE]
640 PRINT"QQQVEGE.
650 END
660 :
670 :
680 :

READY.
```

Tisztelt Szerkesztőség

Bár nem vagyok OCE-tag, és ebből az újságból is csak három darab van (a legutóbbi három), sajnos.

Ezekből az újságokból tudtam meg, hogy ide lehet beküldeni számítógépes programokat. Így én — kapva az alkalmon — csináltam is hármat a C64-re. Ezeket kazettán küldtem el, és nagyon örülnék neki, ha valamelyiket Önök leközölnék a három közül. Ha van kedvük hozzá, akkor talán mind a hármat közreadhatnák (na nem egy lapban).

Ahogy a legutóbbi három számban elnéztem, azt hiszem bármelyikkel felvehetik a versenyt a programjaim.

A programok: TILI-TOLI, HUSZONEGYES, SPRITE KÉSZÍTIŐ

Pozsgai Ferenc

```
0 REM ******************
2 REM *                *
4 REM *  TILI - TOLI   *
6 REM *                *
7 REM ******************
8 :
9 POKE 808,254
10 POKE 657,128:GOSUB 535
15 PRINT CHR$(142)
20 GOTO 245
25 KEY=(PEEK(53266) AND 3)+1
30 SL=(LEPES AND 3):T=LEPES
35 T=T-(KEY=1)-(KEY=3)*4
40 T=T+(KEY=2)+(KEY=4)*4
45 IF T<0 OR T>15 THEN RETURN
50 IF KEY=2 AND SL=0 THEN RETURN
55 IF KEY=1 AND SL=3 THEN RETURN
60 H(LEPES)=H(T)
65 H(T)=15
70 HB=INT(KM(LEPES)/256):LB=KM(LEPES)-
256*HB:POKE 883,LB:POKE 884,HB
75 POKE 880,67
80 SYS 870
85 POKE KM(LEPES)+42,PEEK(KM(T)+42)
90 HB=INT(KM(T)/256):LB=KM(T)-256*HB:
POKE 883,LB:POKE 884,HB
```

```
95 POKE 890,82
100 SYS 870
105 X=0:LEPES=T:RETURN
110 REM  *** KEVERES ***
115 D=60
120 X=1
125 GOSUB 25
130 IF X THEN 125
135 D=D-1:IF D<>0 THEN 120
140 GOSUB 455
145 TI$="000000"
150 POKE 198,0
155 KEY=0
160 IF MOVE THEN 635
165 GET C$:IF C$="" THEN 165
170 IF ASC(C$)=3 THEN 740
175 IF ASC(C$)=29  THEN KEY=1
180 IF ASC(C$)=157 THEN KEY=2
185 IF ASC(C$)=17  THEN KEY=3
190 IF ASC(C$)=145 THEN KEY=4
195 IF KEY=0 THEN 165
200 X=1:GOSUB 30:IF X THEN 155
205 C=1
210 FOR L=0 TO 15
215 IF H(L)<>L THEN C=2
220 NEXT L
225 ON C GOTO 470,155
230 :
235 REM *** GAME START ***
240 :
245 DIM H(15),KM(15)
250 FOR J=835 TO 904
255 READ BYTE
260 CHECK=CHECK+BYTE
265 POKE J,BYTE
270 NEXT
275 IF CHECK<>9442 THEN PRINT"?DATA
ERROR":POKE 808,237:END
280 FOR I=0 TO 15:H(I)=I:NEXT I
285 P=1024+6*40+10:G=P
290 FOR J=0 TO 15
295 KM(J)=P:P=P+5
300 IF P>(G+15) THEN G=G+120:P=G
305 NEXT J
310 :
315 LEPES=15
320 PRINT"▓▓▓";
325 POKE V,15:POKE V+1,15
330 PRINT TAB(6)"  ▌▌▌ ▌▌▌ ▓▓▓▓▓▓▓   ▓▓ ▓▓ ▓▓▓ "
335 PRINT TAB(7)"▌▓▓ ▓ ▓▓ ▓▓   ▓▓▓ ▓▓▓ ▓ ▓ ▓ ▓▓ "
340 PRINT TAB(7)"▌▓▓ ▓  ▓ ▓▓▓▓▓▓▓▓ ▓▓▓▓▓▓ ▓▓▓ ▓  ▓ "
345 PRINT"▓▓▓";
350 PRINT TAB(10)"▓
355 PRINT TAB(10)"▌┌──┐┌──┐┌──┐┌──┐"
360 PRINT TAB(10)"▌│ A ││ B ││ C ││ D │"
365 PRINT TAB(10)"▌└──┘└──┘└──┘└──┘"
370 PRINT TAB(10)"▌┌──┐┌──┐┌──┐┌──┐"
375 PRINT TAB(10)"▌│ E ││ F ││ G ││ H │"
380 PRINT TAB(10)"▌└──┘└──┘└──┘└──┘"
385 PRINT TAB(10)"▌┌──┐┌──┐┌──┐┌──┐"
390 PRINT TAB(10)"▌│ I ││ J ││ K ││ L │"
395 PRINT TAB(10)"▌└──┘└──┘└──┘└──┘"
400 PRINT TAB(10)"▌┌──┐┌──┐┌──┐┌──┐"
405 PRINT TAB(10)"▌│ M ││ N ││ O ││ P │"
410 PRINT TAB(10)"▌└──┘└──┘└──┘└──┘ ▓▓"
415 PRINT"▓▓▓";
420 PRINT TAB(13)"▓             ▓"
425 PRINT TAB(13)"▌  IRANYITAS  ▐"
430 PRINT TAB(13)"▓▌            ▓▓"
435 TI$="000000":GOTO 480
440 PRINT"▓▓▓▓▓";
445 PRINT TAB(14)"MEG KEVEREM."
450 GOTO 115
455 PRINT"▓▓";
460 PRINT TAB(14)"RAKJA SORBA !"
465 RETURN
470 PRINT"▓▓";
475 PRINT TAB(14)"GRATULALOK !!"
480 PRINT"▓▓▓"TAB(14)"TIME "LEFT$(TI$,2)":"MID$(TI$,3,2)":"RIGHT$(TI$,2)
485 PRINT"▓▓▓▓▓"TAB(8)"VALASZTAS UTAN 'RETURN' !!!"
495 GOSUB 680
500 PRINT"▓▓▓▓"TAB(8)"
505 PRINT"▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓▓";
510 PRINT TAB(14)"         "
515 GOTO 440
520 :
525 REM *****       M E N U        *****
530 :
535 V=53280:POKE V,0:POKE V+1,6
540 PRINT"▓▓"CHR$(14);
545 PRINT"▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒";
550 PRINT"▒                                      ▒";
555 PRINT"▒                                      ▒";
560 PRINT"▒                                      ▒";
565 PRINT"▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒"
570 PRINT"▓"
575 PRINT:PRINT:PRINT
580 PRINT"    A GEP KIRAJZOL EGY 4X4-ES MEZOT":PRINT
585 PRINT"    ES ELHELYEZ BENNE 15 BETUT,":PRINT
590 PRINT"    MAJD MEG KEVERI.":PRINT
595 PRINT"    -ELADAT:":PRINT
600 PRINT"    A BETUKET A!--SORRENDBE RAKNI":PRINT
605 PRINT"    AZ URES HELY MOZGATASAVAL.":PRINT:PRINT
610 PRINT"▓▓"TAB(6)"/YOMJON LE EGY BILLENTYUT !"
620 GOSUB 900
625 PRINT"▓";
630 RETURN
635 JOY=PEEK(56321)
640 IF (JOY AND 8)=0 THEN KEY=1
645 IF (JOY AND 4)=0 THEN KEY=2
650 IF (JOY AND 2)=0 THEN KEY=3
655 IF (JOY AND 1)=0 THEN KEY=4
660 IF (JOY AND 16)=0 THEN 740
665 IF KEY=0 THEN 635
```

```
670 GOTO 200
675 :
680 PRINT"S▯▯▯▯▯▯▯▯▯▯▯▯■";:MOVE=0
685 PRINT"F1 - BILL."
690 PRINT"▯▯■F3 - JOY."
695 GET Y$:IF Y$="" THEN 695
700 IF Y$="■" THEN PRINT"TTTT■F1 - BILL.":PRINT"▯▯■F3 - JOY.":MOVE=0
705 IF Y$="■" THEN PRINT"TTTT■F1 - BILL.":PRINT"▯▯■F3 - JOY.":MOVE=1
710 IF ASC(Y$)=13 THEN 720
715 GOTO 695
720 PRINT"TTTT         "
725 PRINT"▯▯             "
730 RETURN
735 :
740 REM *** VEGE ***
745 POKE 198,0:POKE 834,0:SYS 834
750 :
755 DATA 240,195,195,195,238,194,160,160,160,194,237,195,195,195,253
760 DATA 160,160,242,160,160,235,160,160,160,243,160,160,241,160,160,2,2,2,2,2
765 :
770 DATA 160,0,56,162,255,169,5,72,232,185,0,3,157,0,0,200,104,233,1,208,242
775 DATA 24,138,105,35,170,192,15,240,4,56,76,107,3,96
900 :
905 REM *****  SZOVEGMOZGATAS  *****
910 :
915 CREATED$="   T♦♦|♦\  -‾-/-      1992.\\.|.    "
920 TT$="|\L\-|ΓL\   |\L\-|ΓL\   |\L\-|ΓL\    "
925 COL=1
930 POKE 198,0
935 POKE 646,COL+14
940 PRINT"S▯▯▯▯";CREATED$
945 POKE 646,COL+12
950 PRINT"▯▯▯▯▯";TT$
955 CREATED$=RIGHT$(CREATED$,1)+LEFT$(CREATED$,35)
960 TT$=MID$(TT$,2,35)+LEFT$(TT$,1)
965 GET A$:IF A$<>"" THEN RETURN
970 COL=(1-COL)
975 FOR J=0 TO 20:NEXT
980 GOTO 935
999 :
1000 REM   ---   END OF LIST   ---

READY.
```

```
1 REM ------------------------------------------
2 REM -                 - POZSGAI FERENC-
4 REM -   HUSZONEGYES   -   1992.III.01.  -
6 REM -                 -       C=64      -
8 REM ------------------------------------------
9 :
10 DIM CARD$(32),NUMB(32)
15 :
20 POKE 53281,0:POKE 53280,0:PRINT"S▯▯  K E V E R E S":GOSUB 545:GOSUB 410
25 FOR A=1 TO 32 STEP 8
30 RESTORE
35 FOR B=A TO A+7
36 READ CARD$(B)
37 NEXT B,A
40 POKE V,0:POKE V+1,0
45 PRINT"S▯"CHR$(142)" ╭──────────╮"
50 PRINT"|▯JATEKOS▯├──────────────────────────────┤"
55 PRINT"├───────┤                        |"
60 PRINT"|▯PENZ FT▯|                       |"
65 PRINT"|▯"PLAYER"  ▯|                    |"
70 PRINT"├───────┤                        |"
```

```
75 PRINT"|▒  TET   ▒|                                   |"
80 PRINT"|              |                                 |"
85 PRINT"|▒ 0     FT▒|                                    |"
90 PRINT"└──────────┴──────────────────────────────────┘"
95 PRINT"   ▒TALON"
100 FOR A=1 TO 5:PRINT" ░░░░░░░░░░"
105 NEXT A
110 PRINT"▒▒▒▒▒▒▒";
115 PRINT TAB(10)"┌──────────┐ ▒┌──────────┐"
120 PRINT TAB(10)"▒|BANK      |▒|TET       |"
125 PRINT TAB(10)"▒|          |▒|          |"
130 PRINT TAB(10)"▒|"BANK" FT |▒| 0      FT|"
135 PRINT TAB(10)"▒|          |▒|          |"
140 PRINT TAB(10)"▒└──────────┘ ▒└──────────┘"
145 PRINT"▒▒▒▒";
150 PRINT" BANKAR"
155 GOTO 590
165 POKE 198,0
170 PT=0:GOTO 960
171 IF JL=5 AND JER<15 THEN JER=22:GOTO 970
172 IF JL=5 THEN GOTO 730
173 IF BANK=TET THEN 635
174 IF PLAYER=TET THEN 635
175 GET L$:IF L$="" THEN 175
180 J=ASC(L$)
185 IF L$="▒" THEN 215
190 IF L$="▒" THEN 240
205 IF J=13 THEN 330
210 GOTO 175
215 REM *** TET ELORE ***
220 IF (BANK-TET)=PT OR (PLAYER-TET)=PT THEN 175
225 PT=PT+1
230 PRINT"▒▒▒▒▒▒▒▒▒▒▒"PT
235 GOTO 175
240 REM *** TET HATRA ***
245 IF PT=0 THEN 175
250 PT=PT-1
255 PRINT"▒▒▒▒▒▒▒▒▒▒▒▒";
265 PRINT PT"▒ "
270 GOTO 175
325 :
330 REM
335 :
340 TET=TET+PT:PRINT"▒▒▒▒▒▒▒▒▒▒▒TET":PRINT"▒▒▒    ":GOSUB 925:GOTO 635
345 :
350 REM ***** BANKADAS *****
355 REM ***** JATEKOS PENZE *****
360 :
365 BANK=PEEK(53266)+400
370 PLAYER=PEEK(53266)+89
375 RETURN
380 REM **** KARTYAK ****
385 :
400 DATA 2,3,4,7,8,9,10,11
410 :
415 REM *****      START       *****
420 :
425 V=53280
430 PCP=9:JL=0:TET=0
440 GOSUB 350:POKE 650,128
445 TAL=0
450 POKE 657,128:POKE V,254:POKE V+1,254
455 PRINT"▒▒▒";
460 PRINT TAB(11)"▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒"
465 PRINT TAB(11)"▒                  ▒"
470 PRINT TAB(11)"▒▒  |♠♠⌐/⌐| |⌐♠ ▒ ▒"
475 PRINT TAB(11)"▒                  ▒"
480 PRINT TAB(11)"▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒"
485 PRINT"▒▒▒       ♠♠ ⌐| |⌐♠ |\LL⌐/| |/  ⌐L⌐/|⌐♠⌐\▒"
490 PRINT"▒"TAB(13)"-1  TET ELORE▒"
```

```
495 PRINT TAB(13)"-3  TET VISSZA▨"
500 PRINT TAB(13)"'Γ' ONEASZ▨"
505 PRINT TAB(13)"'L' LAPKERES▨"
510 PRINT TAB(13)"'¬' ELDOBAS▨"
515 PRINT "       *Z -1,-3 UTAN '_⌐| /-/ KELL"
520 PRINT "     ¯ZUTAN A '|¯|'-FELIRAT INVERZ LESZ"
522 PRINT "       ES JELZI,HOGY HASZNALHATJUK AZ"
523 PRINT "            (Γ,L,¬) BILLENTYUKET.▨"
525 PRINT TAB(7)"▨/YOMJON LE EGY BILLENTYUT !";
530 POKE 198,0
535 GET B$:IF B$="" THEN 535
540 RETURN
545 FOR I=1 TO 32
550 RAN=(PEEK(53266) AND 31)+1
555 FOR J=1 TO I
560 IF NUMB(J)=RAN THEN 550
565 NEXT J
570 NUMB(I)=RAN
575 NEXT I
585 RETURN
590 :
595 REM *** OSZTAS ***
600 PRINT"▨▨▨▨▨▨▨▨      OSZTAS A JATEKOSNAK       "
610 :
611 KK=NUMB(TAL+1)/8
612 IF ASZ=1 AND INT(KK)=KK THEN TAL=TAL+1:GOTO 611
613 IF ASZ=1 THEN ASZ=0:PRINT"▨"TAB(17)"        "
615 TAL=TAL+1:JL=JL+1
620 KH=INT(NUMB(TAL)/8.3):GOSUB 745
625 ON KH+1 GOSUB 785,815,845,885
630 PCP=PCP+6:IF TET<>BANK AND TET<>PLAYER THEN PRINT"▨▨▨▨▨▨▨▨▨▨TET"
631 GOTO 165
635 GET X$:IF X$="" THEN 635
640 IF X$="O" THEN 660
645 IF X$="L" THEN 700
650 IF X$="D" AND PT=0 THEN 710
655 GOTO 635
660 REM *** ONEASZ ***
665 IF JL=1 THEN 635
675 T=LSZ(1)+LSZ(2)+LSZ(3)+LSZ(4)
680 IF T<>11 THEN 635
685 PRINT"▨▨"TAB(17)"ONEASZ"
690 ASZ=1
695 GOTO 635
700 REM ***  LAPKERES ***
705 GOTO 610
710 REM *** ELDOBAS ***
715 IF JL=1 THEN 635
720 JER=LSZ(1)+LSZ(2)+LSZ(3)+LSZ(4)
725 IF JER<15 THEN 635
730 PRINT"▨▨"TAB(17)"BEDOBVA"
740 GOTO 980
745 REM
750 :
755 PRINT"▨▨▨▨";
760 R=NUMB(TAL)
765 D$=LEFT$(CARD$(R),2):F$=D$
770 LSZ(JL)=VAL(D$):D$=F$+"   "+F$
775 IF F$="10" THEN D$="10 10"
776 IF F$="11" THEN D$="A    A"
780 RETURN
785 REM *** PIKK ***
790 PRINT TAB(PCP)"▨"D$
795 FOR U=1 TO 5:PRINT TAB(PCP)"▨  ♠  "
800 NEXT U
805 PRINT TAB(PCP)"▨"D$
810 RETURN
815 REM *** KOR ***
820 PRINT TAB(PCP)"▨"D$
825 FOR U=1 TO 5:PRINT TAB(PCP)"▨ ♥ ♥ "
```

```
830 NEXT U
835 PRINT TAB(PCP)"▒"D$
840 RETURN
845 REM *** KARO ***
850 PRINT TAB(PCP)"▒"D$
855 PRINT TAB(PCP)"▒♦    ♦"
860 FOR U=1 TO 3:PRINT TAB(PCP)"▒     "
865 NEXT U
870 PRINT TAB(PCP)"▒♦    ♦"
875 PRINT TAB(PCP)"▒"D$
880 RETURN
885 REM *** TREFF ***
890 PRINT TAB(PCP)"▒"D$
895 PRINT TAB(PCP)"▒  ♣   ":PRINT TAB(PCP)"▒
900 PRINT TAB(PCP)"▒♣    ♣":PRINT TAB(PCP)"▒
905 PRINT TAB(PCP)"▒  ♣   "
910 PRINT TAB(PCP)"▒"D$
915 RETURN
920 :
925 REM
930 PRINT"▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒";
935 PRINT TAB(22)TET"▒  "
955 RETURN
960 REM
965 JER=LSZ(1)+LSZ(2)+LSZ(3)+LSZ(4)+LSZ(5)
970 IF JER>21 AND JL<>2 THEN PRINT"▒▒"TAB(17)"FUCCS":Y=1:GEP=1:GOTO 1130
975 GOTO 171
980 REM *** BANKOS LAPJAI ***
985 :
990 PRINT"▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒";
995 PRINT"                              "
1000 PCP=9:JL=5
1010 :
1011 KK=NUMB(TAL+1)/8
1012 IF ASZ=1 AND KK=INT(KK) THEN TAL=TAL+1:GOTO 1011
1013 IF ASZ=1 THEN ASZ=0
1015 TAL=TAL+1:JL=JL+1
1020 KH=INT(NUMB(TAL)/8.3)
1025 PRINT"▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒";
1030 GOSUB 760
1035 ON KH+1 GOSUB 785,815,845,885
1040 PCP=PCP+6
1045 OA=LSZ(6)+LSZ(7)+LSZ(8)+LSZ(9)+LSZ(10)
1046 IF JL=10 AND OA<15 THEN OA=22:GOTO 1080
1047 IF JL=10 THEN 1080
1050 IF JL<>6 AND OA=11 THEN ASZ=1
1055 IF OA<15 THEN 1010
1060 IF LSZ(6)=11 AND LSZ(7)=11 THEN GEP=1:GOTO 1130
1065 IF OA>19 THEN 1080
1070 BLAP=INT(RND(1)*(OA-13))
1075 IF BLAP=0 THEN 1010
1080 IF OA>21 THEN PRINT TAB(17)"▒FUCCS";:GEP=0:GOTO 1130
1085 PRINT TAB(17)"▒BEDOBVA";
1100 IF JER=22 THEN GEP=0:GOTO 1130
1120 IF OA>=JER THEN GEP=1
1125 IF OA<JER THEN GEP=0
1130 IF GEP=1 THEN PLAYER=PLAYER-TET:BANK=BANK+TET
1135 IF GEP=0 THEN PLAYER=PLAYER+TET:BANK=BANK-TET
1145 PRINT"▒▒▒▒▒▒▒"PLAYER"▒  "
1170 PRINT"▒▒▒▒▒▒▒▒▒▒";
1175 PRINT TAB(11)BANK"▒  "
1176 POKE 198,0
1177 GET E$:IF E$="" THEN 1177
1180 PRINT"▒"TAB(17)"          "
1190 PRINT"▒▒▒▒▒▒▒▒▒▒▒▒▒▒"TAB(23)"▒▒   ▒▒▒▒▒▒▒▒▒▒"
1195 PRINT TAB(17)"          ";
1200 PRINT"▒▒▒";
1205 PCP=9:JL=0:TET=0
1207 FOR I=1 TO 10:LSZ(I)=0:NEXT I
1210 IF BANK=0 THEN 1300
```

```
1215 IF PLAYER=0 THEN 1495
1220 FOR L=1 TO 7
1225 PRINT TAB(9)"                    ":NEXT L
1230 IF Y=1 THEN 1255
1235 PRINT"▒▒▒▒▒▒▒▒";
1240 FOR L=1 TO 7:PRINT TAB(9)"
1245 NEXT L
1250 PRINT"]]]]]]]]     ▒        OSZTAS A JATEKOSNAK       "
1255 Y=0:IF TAL<24 THEN 610
1260 PRINT"▒▒▒▒▒▒▒▒▒▒▒▒";:FOR J=1 TO 32:NUMB(J)=0:NEXT J
1265 PRINT" ▒▒KEVERES":TAL=0
1270 GOSUB 545
1275 PRINT"] ▒TALON▒ "
1280 GOTO 610
1290 :
1300 REM ***** BANKROBBANTAS *****
1305 PRINT"▒▒▒▒▒";
1310 PRINT TAB(9)"●●●    ●●   ●   ●  ●  ●"
1315 PRINT TAB(9)"●  ●  ●  ●  ●●  ●  ●  ●"
1320 PRINT TAB(9)"●●●   ●●●●  ● ● ●  ●●"
1325 PRINT TAB(9)"●  ●  ●  ●  ●  ●●  ● ●"
1330 PRINT TAB(9)"●●●   ●  ●  ●  ●  ●  ●▒▒▒"
1335 PRINT"●●●   ●●  ●●●  ●●●   ●●  ●   ●  ●●    ●●"
1340 PRINT"●  ●  ●  ●  ●  ●  ●  ●  ●  ●●  ●  ●  ●  ●"
1345 PRINT"●●●,  ●   ●  ●●●  ●●●   ●●●● ● ● ● ●●●●  ●●"
1350 PRINT"●  ●   ●   ●  ●  ●  ●  ●  ●  ●  ●●  ●  ●  ●"
1355 PRINT"●  ●  ●●  ●●●  ●●●  ●  ●  ●  ●  ●  ●  ●●●▒▒▒▒"
1360 T=8
1365 FOR P=1 TO 23
1370 PRINT TAB(T)MID$("MEG EGY JATSZMA  (I/N)",P,1);
1375 FOR R=0 TO 100:NEXT R
1380 T=T+1
1385 NEXT P
1390 POKE 198,0
1395 GET T$:IF T$="" THEN 1395
1400 IF T$="N" THEN SYS 2048
1405 IF T$="I" THEN 1415
1410 GOTO 1395
1415 TAL=0:Y=0
1420 FOR I=1 TO 32:NUMB(I)=0:NEXT I
1425 GOSUB 1570
1430 GOSUB 545
1435 GOSUB 350
1440 GOTO 45
1490 :
1495 REM ***** JATEKOS VESZTETT *****
1500 :
1505 PRINT"▒▒▒▒▒";
1510 PRINT TAB(7)"●   ●   ●   ●   ●   ●●   ●●"
1515 PRINT TAB(7)"●●   ●   ●   ●●   ●   ●   ●"
1520 PRINT TAB(7)"● ●  ●  ●   ● ●  ●  ●    ●●"
1525 PRINT TAB(7)"●   ●●   ●   ●   ●●   ●    ●"
1530 PRINT TAB(7)"●   ●   ●   ●   ●   ●●   ●●●▒▒▒"
1535 PRINT TAB(8)"●●●   ●●●  ●   ●  ●●●●  ●●●"
1540 PRINT TAB(8)"●   ●  ●   ●●   ●   ●  ●"
1545 PRINT TAB(8)"●●●   ●●   ●  ●  ●   ●   ●●"
1550 PRINT TAB(8)"●   ●   ●   ●●   ●●   ●"
1555 PRINT TAB(8)"●   ●●●  ●   ●  ●●●●  ●●●▒▒▒▒"
1560 GOTO 1360
1570 PRINT"▒▒▒▒▒▒▒▒▒▒";
1575 PRINT"        ▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒"
1580 PRINT"        ▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒"
1585 PRINT"        ▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒"
1590 PRINT"        ▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒"
1595 PRINT"        ▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒"
1600 RETURN

READY.
```

```
10 REM ******************************
15 REM *****  SPRITE-KESZITO   *****
20 REM *****                  *****
35 REM *****  POZSGAI FERENC  *****
40 REM *****                  *****
45 REM *****    1992.II.10.   *****
50 REM ******************************
55 :
60 POKE 657,128:PRINT CHR$(142)
65 DIM BS(23)
70 GOSUB 505
75 PNT=0:BYTE=0:Q=0:W=0:SP=219
80 CM=55389:KM=1117
85 V=53248
90 POKE V+21,0
95 POKE V+40,1:POKE V+16,0
100 POKE 2041,13
110 FOR N=0 TO 63:POKE 832+N,0:NEXT N
115 :
120 PRINT"█"TAB(18)"█SPRITE-KESZITO██"
125 FOR J=1 TO 21
130 PRINT TAB(13)"█                        ":NEXT
131 PRINT"█";TAB(15)"SPRITE KIKAPCSOLVA";
135 POKE KM,SP
140 :
145 POKE 198,0
150 GET KEY$:IF KEY$="" THEN 150
155 IF KEY$="█" AND PNT<>23 THEN 225
160 IF KEY$="█" AND PNT<>0 THEN 245
165 IF KEY$="█" AND BYTE<>60 THEN 265
170 IF KEY$="█" AND BYTE<>0 THEN 285
175 IF KEY$="█" THEN 305
180 IF KEY$="█" THEN 365
185 IF KEY$="█" THEN GOSUB 390
190 IF KEY$="█" THEN GOSUB 455:GOSUB 475:GOTO 110
195 IF KEY$="█" AND PEEK(V+21)=2 THEN 415
200 IF KEY$="█" AND PEEK(V+21)=2 THEN 435
205 IF KEY$="█" THEN GOSUB 455
210 IF KEY$="█" THEN 800
220 GOTO 145
221 :
222 :
223 REM
224 :
225 REM ***     JOBBRA      ***
226 :
227 REM
228 :
229 :
230 KM=KM+1:PNT=PNT+1:CM=CM+1
235 POKE KM-1,160:POKE KM,SP
240 GOTO 150
241 :
242 :
243 REM
244 :
245 REM ***     BALRA      ***
246 :
247 REM
248 :
249 :
250 KM=KM-1:PNT=PNT-1:CM=CM-1
255 POKE KM+1,160:POKE KM,SP
260 GOTO 150
261 :
262 :
263 REM
264 :
265 REM ***     LE      ***
266 :
267 REM
268 :
269 :
270 KM=KM+40:BYTE=BYTE+3:CM=CM+40
275 POKE KM-40,160:POKE KM,SP
280 GOTO 150
281 :
282 :
283 REM
284 :
285 REM ***      FEL      ***
286 :
287 REM
288 :
289 :
290 KM=KM-40:BYTE=BYTE-3:CM=CM-40
295 POKE KM+40,160:POKE KM,SP
300 GOTO 150
304 REM
305 REM ***      PONT BE      ***
306 REM
310 MASK=832+BYTE+INT(PNT/8)
315 IF (PEEK(MASK)AND BS(PNT)) THEN 335
320 POKE CM,1
325 POKE MASK,PEEK(MASK)OR BS(PNT)
330 GOTO 150
334 REM
335 REM ***      PONT KI      ***
336 REM
345 POKE CM,14
355 POKE MASK,PEEK(MASK)AND 255-BS(PNT)
360 GOTO 150
361 :
365 REM ***  SPRITE-ADATOK KIIRASA  ***
369 :
370 PRINT"████";
```

```
375 FOR A=832 TO 892 STEP 3
380 PRINT "▓             ":PRINT "T";PEEK(A);"▓";PEEK(A+1);"▓";PEEK(A+2)
385 NEXT A:GOTO 145
387 :
388 :
390 :         REM *** SPRITE BE ***
392 :
393 :
394 IF PEEK(V+21)=2 THEN 480
395 X=48:Y=114
400 POKE V+2,X:POKE V+3,Y
405 POKE V+21,2
406 POKE 2006,2:POKE 2007,5
410 RETURN
415 REM *  X-NAGYITAS / X-ZSUGORITAS  *
420 Q=(2-Q)
425 POKE V+29,Q
430 GOTO 150
435 REM *  Y-NAGYITAS / Y-ZSUGORITAS  *
440 W=(2-W)
445 POKE V+23,W
450 GOTO 150
451 :
455 REM ** SPRITE-ADATOK TORLESE   **
459 :
460 PRINT"▓▓▓";
465 FOR J=1 TO 21:PRINT"                         "
470 NEXT J:RETURN
472 :
473 :
475 :    REM ***     SPRITE KI       ***
477 :
478 :
480 POKE V+21,0
485 Q=0:W=0
490 POKE V+23,0:POKE V+29,0
491 POKE 2006,11:POKE 2007,9
495 RETURN
500 :
505 REM **********   MENU   **********
510 :
515 PRINT"▓":FOR C=0 TO 23:READ BS(C):NEXT C
520 POKE 53281,0:POKE 53280,0
525 PRINT TAB(13)"▓▓SPRITE-KESZITO"
530 PRINT"▓▓PARANCSOK:"
535 PRINT"▓▓▓"TAB(8)"F1 - PONT KI/BE"
540 PRINT TAB(8)"▓F3 - SPRITE-ADATOK KIIRASA"
545 PRINT TAB(8)"▓F5 - SPRITE LATSZIK/NEM LATSZIK"
550 PRINT TAB(8)"▓F7 - SPRITE TORLESE"
555 PRINT TAB(8)"▓F2 - X-NAGYITAS / X-KICSINYITES"
560 PRINT TAB(8)"▓F4 - Y-NAGYITAS / Y-KICSINYITES"
565 PRINT TAB(8)"▓F6 - SPRITE-ADATOK TORLESE"
570 PRINT TAB(8)"▓F8 - VEGE"
575 PRINT TAB(8)"▓▓▓NYOMJON LE EGY BILLENTYUT";
580 POKE 198,0
585 GET D$:IF D$="" THEN 585
590 PRINT"▓"
595 RETURN
600 :
605 :
610 :
700 DATA 128,64,32,16,8,4,2,1,128,64,32,16,8,4,2,1,128,64,32,16,8,4,2,1
705 :
715 :
745 :
800 POKE 255,0
805 SYS 255

READY.
```

SPRITE Lopó

E program segítségével a gép memóriájában lévő sprite-okat lehet lelopni. A programmal megnyílik a lehetőség, hogy a játékprogramok nagyon jól megtervezett SPRITE-jait megszerezzük magunknak is.

A rendszer létrehozása és betöltése:

1. Futtassuk le a SPRITE THIEF programot, ami lényegében DATA OLVASÓ. Ebben található két program DATA sorokban. Névszerint: LOADER (SPRITE): a rendszer betöltése; SPRITE: maga a SPRITE Lopó.

A futtatás után ha mindent jól írtunk be, akkor az előbb említett két program a lemezen lesz, és a gép végrehajtja a RESET rutint.

2. Töltsük be a ,,lelopandó'' programot és RESET. Ha dolgozni akarunk a programmal, akkor töltsük be a LOADER-t LOAD ″LOADER (SPRITE)″,8,8

Ezután a SPRITE THIEF-ben vagyunk. Itt nyílik lehetőség a SPRITE-ok megkeresésére. A rendszer használata:

Billentyűk:

'+' Egy SPRITE-tal előre lép a memóriában.
'–' Egy SPRITE-tal visszalép a memóriában.
'Balra nyíl' 16 SPRITE-tal előre lép a memóriában.
'S' SPRITE kimentése.

Kilépés a programból 'SPACE'

'F1' A SPRITE 1. színének a megváltoztatása.
'F2' A SPRITE 2. színének a megváltoztatása.
'F5' A SPRITE 3. színének megváltoztatása.
'M' MULTI-COLOR mód.
'H' HIRES mód.

SAVE:

Az 'S' megnyomása után a program kiírja: PRESS ANY KEY

Az itt megnyomott betű lesz a SPRITE neve a lemezen. A név formátuma: SPRx.

Egyéb tudnivalók

1. A programból kilépni csak a SPACE-szel vagy RESET-tel lehet, mert a program átírja a $302-es vektort, és ez mindig loaderre mutat. Ha megnyomjuk a RUN/STOP + RESTORE-t, akkor a betöltést újra elindítjuk, és az keresi a rendszert a lemezen, ha nem találja meg, akkor a gép befagy.

2. Az 'S'-el kimentett SPRITE $400 (1024) címre fog betölteni, de ezt a címet egy LOAD ADDRESS programmal át lehet írni bármire.

3. Ha a helyére betöltünk egy SPRITE-ot, akkor megjeleníteni a következőképpen lehet:

POKE 2040,16 (SPRITE helye a memóriában)
POKE 53248,100 SPRITE x koordinátája
POKE 53249,100 SPRITE y koordinátája
POKE 53269,1 SPRITE bekapcsolása.

Ezzel megjelenítettük a SPRITE-ot. Ha MULTI-COLOR (többszínű) módban szeretnénk látni, akkor még egy POKE-ot be kell írni:

POKE 53276,1

A négyszeres nagyítást két POKE segítségével érthetjük el.

POKE 53271,1 x irányú nagyítás
POKE 53277,1 y irányú nagyítás.

4. A rendszer betöltve 3 BLOCK-ot foglal el a lemezen. Egy SPRITE 1 BLOCK-ot.

5. Ha két SPRITE-ot ugyanazzal a névvel akarunk kimenteni, akkor a floppy-n a led elkezd villogni.

Gyárfás Richárd

```
10 REM ********************************
20 REM * SPRITE THIEF BY DARK VISION *
30 REM *                  SOFTWARE. *
40 REM *                      (DVS)  *
50 REM * AGLOSOFT  PRESENT           *
60 REM ********************************
70 REM
80 OPEN2,8,2,"@:SPRITE,P,W"
90 PRINT#2,CHR$(9*16)CHR$(5);
100 READA:IFA=-1THEN130
110 S=S+A
120 PRINT#2,CHR$(A);:GOTO100
130 CLOSE2:IFS=59320THEN160
140 OPEN15,8,15,"S:SPRITE":CLOSE15
150 PRINT"▮DATA ERROR (PROGRAM)":STOP
160 GOTO780
170 DATA 169,0,141,32,208,141,33,208
180 DATA 169,12,141,134,2,169,1,141
190 DATA 23,208,141,29,208,76,187,5
200 DATA 160,0,177,34,201,13,240,7
210 DATA 32,71,171,200,76,170,5,32
220 DATA 71,171,96,169,16,141,248,7
230 DATA 169,1,141,21,208,169,160,141
240 DATA 0,208,169,100,141,1,208,169
250 DATA 0,133,249,133,250,32,94,6
260 DATA 32,159,255,32,228,255,201,0
270 DATA 240,246,160,160,162,0,232,208
280 DATA 253,200,208,248,162,128,142,138
290 DATA 2,201,0,240,227,162,0,134
300 DATA 198,201,43,240,65,201,45,240
310 DATA 77,201,95,240,51,201,77,240
320 DATA 32,201,72,240,31,201,83,240
330 DATA 42,201,133,240,26,201,134,240
340 DATA 25,201,135,240,24,201,32,208
350 DATA 183,169,0,141,21,208,76,226
360 DATA 252,76,151,6,76,159,6,76
370 DATA 167,6,76,178,6,76,184,6
380 DATA 76,140,6,76,209,6,165,249
390 DATA 24,105,64,133,249,165,250,105
400 DATA 0,133,250,76,94,6,165,249
410 DATA 56,233,64,133,249,165,250,233
420 DATA 0,133,250,76,94,6,32,55
430 DATA 7,160,64,120,169,53,133,1
440 DATA 177,249,162,55,134,1,88,153
450 DATA 0,4,136,192,255,208,236,169
460 DATA 3,133,214,169,0,133,211,32
470 DATA 108,229,165,250,166,249,32,205
480 DATA 189,76,216,5,230,250,230,250
490 DATA 230,250,230,250,76,94,6,169
500 DATA 255,141,28,208,76,216,5,169
510 DATA 0,141,28,208,76,216,5,238
520 DATA 39,208,169,32,141,138,2,76
530 DATA 216,5,238,38,208,76,170,6
540 DATA 238,37,208,76,170,6,80,82
550 DATA 69,83,83,32,65,78,89,32
560 DATA 75,69,89,13,83,65,86,69
570 DATA 13,169,0,133,198,169,4,133
580 DATA 214,169,0,133,211,32,16,229
590 DATA 162,6,134,35,169,190,133,34
600 DATA 32,168,5,169,32,141,138,2
610 DATA 169,0,133,198,32,159,255,32
620 DATA 228,255,201,0,240,246,141,63
630 DATA 3,169,83,141,60,3,169,80
640 DATA 141,61,3,169,82,141,62,3
650 DATA 169,0,141,64,3,162,8,32
660 DATA 186,255,169,4,162,60,160,3
670 DATA 32,189,255,169,0,133,251,169
680 DATA 4,133,252,160,4,169,251,162
690 DATA 65,32,216,255,76,94,6,32
700 DATA 91,7,162,0,169,32,157,0
710 DATA 4,169,1,157,0,216,232,208
720 DATA 243,162,0,169,32,157,0,5
730 DATA 169,1,157,0,217,232,224,144
740 DATA 208,241,96,169,0,133,34,162
750 DATA 216,134,35,160,0,145,34,230
760 DATA 34,208,250,230,35,166,35,224
770 DATA 220,208,242,96,-1
780 S=.:U=828
790 READA:IFA=-1THEN810
800 POKEU,A:U=U+1:S=S+A:GOTO790
810 IFS<>3356THENPRINT"▮ATA ERROR (LOADER)":STOP
820 U=3*256+2:POKEU,60:POKEU+1,3
830 OPEN2,8,2,"@:LOADER (SPRITE),P,W"
840 PRINT#2,CHR$(2)CHR$(3);
850 FORI=UTO862
860 PRINT#2,CHR$(PEEK(I));:NEXT:CLOSE2:SYS64738
870 DATA 76,69,3,83,80,82,73,84
880 DATA 69,162,8,160,1,32,186,255
890 DATA 169,6,162,63,160,3,32,189
900 DATA 255,169,0,32,213,255,76,144
910 DATA 5,-1

READY.
```

```
100 REM ******************************
110 REM *   COLOUR BASIC V1.0   C-64*
120 REM *   ==================      *
130 REM *  ,  ,                     *
140 REM * KESZITETTE: BRUCE MAESTRO *
150 REM ******************************
160 :
170 ::FOR I= 49152 TO 49563
180 :::PRINT"█";I
190 :::READA:POKEI,A:S=S+A
200 ::NEXT
210 :IF S<> 49819 THEN PRINT"HIBA"
220 DATA 169,011,141,008,003,169,192
230 DATA 141,009,003,096,032,115,000
240 DATA 201,033,240,006,032,121,000
250 DATA 076,231,167,032,115,000,201
260 DATA 075,208,003,032,141,192,201
270 DATA 072,208,003,032,159,192,201
280 DATA 065,208,003,032,177,192,201
290 DATA 066,208,003,032,195,192,201
300 DATA 067,208,003,032,213,192,201
310 DATA 068,208,003,032,231,192,234
320 DATA 234,234,234,234,201,069,208
330 DATA 003,032,249,192,201,048,208
340 DATA 003,032,011,193,201,049,208
350 DATA 003,032,029,193,201,050,208
360 DATA 003,032,047,193,201,051,208
370 DATA 003,032,065,193,201,052,208
380 DATA 003,032,083,193,201,053,208
390 DATA 003,032,101,193,201,052,208
400 DATA 003032,119,193,201,054,208
410 DATA 003,032,137,193,076,008,175
420 DATA 255,032,115,000,032,253,174
430 DATA 032,158,183,180,000,142,032
440 DATA 208,076,174,167,255,032,115
450 DATA 000,032,253,174,032,158,183
460 DATA 180,000,142,033,208,076,174
470 DATA 167,255,032,115,000,032,253
480 DATA 174,032,158,183,180,000,142
490 DATA 034,208,076,174,167,255,032
500 DATA 115,000,032,253,174,032,158
510 DATA 183,180,000,142,035,208,076
520 DATA 174,167,255,032,115,000,032
530 DATA 253,174,032,158,183,180,000
540 DATA 142,036,208,076,174,167,255
550 DATA 032,115,000,032,253,174,032
560 DATA 158,183,180,000,142,037,208
570 DATA 076,174,167,255,032,115,000
580 DATA 032,253,174,032,158,183,180
590 DATA 000,142,038,208,076,174,167
600 DATA 255,032,115,000,032,253,174
610 DATA 032,158,183,180,000,142,039
620 DATA 208,076,174,167,255,032,115
630 DATA 000,032,253,174,032,158,183
640 DATA 180,000,142,040,208,076,174
650 DATA 167,255,032,115,000,032,253
660 DATA 174,032,158,183,180,000,142
670 DATA 041,208,076,174,167,255,032
680 DATA 115,000,032,253,174,032,158
690 DATA 183,180,000,142,042,208,076
700 DATA 174,167,255,032,115,000,032
710 DATA 253,174,032,158,183,180,000
720 DATA 142,043,208,076,174,167,255
730 DATA 032,115,000,032,253,174,032
740 DATA 158,183,180,000,142,044,208
750 DATA 076,174,167,255,032,115,000
760 DATA 032,253,174,032,158,183,180
770 DATA 000,142,045,208,076,174,167
780 DATA 255,032,115,000,032,253,174
790 DATA 032,158,183,180,000,142,046
800 DATA 208,076,174,167,255,096
810 SYS49152
```

Colour Basic

A Colour Basic-ban utasítások sorozatával állíthatjuk gépünk színeit.

!A,X: többszínű üzemmód 1. színét állíthatjuk be.
!B,X: többszínű üzemmód 2. színét állíthatjuk be.
!C,X: többszínű üzemmód 3. színét állíthatjuk be.
!D,X: többszínű sprite 1. színe.
!E,X: többszínű sprite 2. színe.
!K,X: keret szín.
!H,X: háttér szín.
!O,X: 0. sprite színe.
!1,X: 1. sprite színe.
!2,X: 2. sprite színe.
!3,X: 3. sprite színe.
!4,X: 4. sprite színe.
!5,X: 5. sprite színe.
!6,X: 6. sprite színe.

Az X koordinátában azt kell beállítani, hogy milyen színt akarunk a spritenek, háttérnek stb.

```
820 PRINTCHR$(147):!K,0:!H,0:POKE646,5
830 PRINTTAB(6)"*** COLOUR BASIC V1.0 ***
840 PRINTTAB(16)"BY
850 PRINTTAB(11)"BRUCE MAESTRO
860 PRINT"█UTMUTATAST [I/N]
870 GETA$
880 IFA$="I"THENGOTO910
890 IFA$="N"THENEND
900 GOTO870
910 PRINT"█UTMUTATAS :
████████████▔▔▔▔▔▔▔▔▔▔▔▔▔▔":E=0
920 PRINT"█████████UJ UTASITASOK,PARANCSOK :
930 PRINT"████████▔▔▔▔▔▔▔▔▔▔▔▔▔▔▔▔▔▔▔▔▔▔▔
940 PRINT"█!A ...... TOBBSZINU UZEMMOD 1.SZIN
950 PRINT"!B ...... TOBBSZINU UZEMMOD 2.SZIN
960 PRINT"!C ...... TOBBSZINU UZEMMOD 3.SZIN
970 PRINT"!D ...... TOBBSZINU SPRITE 1.SZIN
980 PRINT"!E ...... TOBBSZINU SPRITE 2.SZIN
990 PRINT"!K ...... KERET SZIN
1000 PRINT"!H ...... HATTER SZIN
1010 PRINT"!0 ...... 0. SPRITE SZINE
1020 PRINT"!1 ...... 1. SPRITE SZINE
1030 PRINT"!2 ...... 2. SPRITE SZINE
1040 PRINT"!3 ...... 3. SPRITE SZINE
1050 PRINT"!4 ...... 4. SPRITE SZINE
1060 PRINT"!5 ...... 6. SPRITE SZINE
1070 PRINT"!6 ...... 5. SPRITE SZINE
1080 PRINTTAB(32)"██<SPACE>":POKE198,0:WAIT198,1
1090 !K,0:!H,0:FORI=0TO100:NEXT:E=E+1:!K,1:!H,1:
FORI=0TO100:NEXT
1100 IFE=10THENPOKE53280,0:!H,0:GOTO1120
1110 GOTO1090
1120 POKE198,0:WAIT198,1:GOTO820
1130 ***** ITT A VEGE *****
```

ORBIT F

Összenyomóprogram C + 4-re

Manapság igen elterjedtek a különféle elven működő összenyomó-, másnéven compacker vagy tömörítőprogramok. Hogy csak néhányat említsek: a legáltalánosabban használt ilyen tömörítők bájtos elven működnek, rendszerint ezeknek legnagyobb a hatásfokuk. Utánuk következnek a bitsorozatokat kereső és egyéb nyakatekert elvű compackerek, amelyeknek hatásfoka már jóval kisebb, megéri azonban ezeket használni, mert a már bájtos elven egyszer összetömörített programokat is képesek tovább gyömöszölni. Ebbe a kategóriába tartoznak egyébként a HCS-packer és változatai.

A harmadik ,,generációs'' összenyomók már elég keveset nyomnak a programon, mégis a HCS-típusú tömörítők nyomait is összébb préselik. Ezek már többnyire statisztikai elven működnek. Az itt bemutatott ORBIT F nevű program is ilyen.

A DATA sorok első látásra soknak tűnnek, ám a helyzet korántsem ilyen vészes. Ha begépeltük az összes sort, még elindítás előtt feltétlenül mentsük ki a BASIC nyelvű programot, mert ennyi karakter begépelésekor annak valószínűsége, hogy valamit félreütöttünk, elég magas. Hibátlan bevitel esetén elindításkor a BASIC lassúsága miatt 3—4 percet kell várnunk a gépi kódú sorok bevitelére. Ez idő alatt a $1000—$2830-ig terjedő memóriaterületre kerülnek bájtok, így ha a begépelést ,,tiszta lappal'' ($1001-es BASIC kezdettel) kezdtük, a DATA sorok felülíródnak, s ha ezután a program felhívja a figyelmünket az egyik sor adathibájára, azt csak az előzőleg kimentett programon tudjuk kijavítani.

S ha végül a BASIC program azt üzeni, hogy készen van, s kiírja az ORBIT F üzenetet, azonnal írjunk be egy BASIC SAVE parancsot, hogy a gépi kódú részt kimentsük. A továbbiakban a BASIC DATA sorokra már nem lesz szükség, elegendő a gépi kódú programot betölteni, és RUN-nal futtatni. (Aki monitorban szeret bütykörészni, annak elmondom, hogy a program kezdő- és végcímét a $2B—$2E bájtokban találja ($1001—$2826).)

Elindítás után megjelenik a címképernyő: ez tömören elmondja, mi is a program lényege, erről kissé bővebben a későbbiekben szólok. Majd egy tömör ismertetőt találunk a billentyűkről és a program által használt nulláslapos címekről. Ha kigyönyörködtük magunkat, nyomjuk le egyszerre a SHIFT és a CONTROL billentyűket, ekkor hamarosan megjelenik a képernyő tetején a READY felirat, alatta pedig egy csomó krikszkraksz. Nem kell megijedni, ez maga a program, úgyhogy lehetőleg ne nagyon nyomkodjuk az ESCAPE-et vagy kétszer a HOME-ot, hacsak nem akarunk búcsút venni a programtól.

Akkor hadd szóljak most a program kezeléséről: ha kazettával dolgozunk, az F3 billentyűvel keresgélhetünk, a SHIFT megnyomásával visszakapjuk a képet. Ezután a HELP betölti az első talált programot. Lemezeseknek sajnos kézzel kell beírniuk a DLOAD parancsot, de ezért bőségesen kárpótol, amit a program tud. F7-tel kilistázhatjuk a betöltött programot. Figyelem: ne nagyon használjunk olyan gyorstöltőket, amelyek a $0400—04FF és $0600—06FF területeket használják!

Ha értelmesen akarjuk használni az ORBIT F-et, az összenyomandó programot előzőleg vessük alá egy bájtos és egy HCS

```
0 goto63000
1 data 00191037099e343239333a1b50,27e
2 data 204f5242495420462021210000,268
3 data 000000000000a9008d19ff8d07,2e2
4 data ffa91b8d06ffa9718d3b05a919,5fe
5 data 8d15ffa9c08d12ffa9148d13ff,704
6 data 2088d8a90085d2a918eaa92785,680
7 data d0a90c85d1a91885d3a90185d4,6f7
8 data a000b1d291d0c8c4d4d0f7e6d4,965
9 data a0e8a2e0e8d0fdc8d0f8a5d0d0,a94
10 data 02c6d1c6d0ce0fffc6d5d0daa5,8f5
11 data d018695085d09002e6d1a5d218,6ce
12 data 692885d29002e6d3a5d1c910f0,772
13 data 0ba9d08d0fffa92885d5d0af4c,715
14 data e0100000000000000000000000,0f0
15 data 00000000000000000000000000,000
16 data 0000a92885d5a9488d11ffa9d0,632
17 data 8d0fffad10ff09038d10ff4c20,56b
18 data 100000a00884d0ce0fffa2d0a0,5fa
19 data d0c8d0fde8d0f8c6d0d0efce11,a49
20 data ffad11ff2907d0e1a9008d4305,61b
21 data ad4305c905d0f978a90085d0a9,6ab
22 data cc85d1a9008d1403a9128d1503,4cf
23 data 58a5d0c966d0fa2088d8a90385,777
24 data d0a200a000c8d0fde8d0f8c6d0,8ed
25 data d0f2a200bd00209d0006bd0021,4c2
26 data 9d0009bd00229d000abd00239d,3a9
27 data 000bbd00249d000dbd00259d00,315
28 data 0ebd00269d000fe8d0d3a20fbd,596
29 data 00279df008bd10279df00cca10,523
30 data f1a25dbd00279d0005e8d0f7a9,6ce
31 data c48d12ffa9d08d13ffa9418d19,70a
32 data ffa9718d15ffa9008d11ffa905,6ae
33 data 8de507a9008de6078de707a927,5e7
34 data 8de807a9008d3b05a9088d07ff,536
35 data 58a225bd00289d0004ca10f74c,4c2
36 data 03870000000000000000000000,08a
37 data 00000000000000000000000000,000
38 data 00000000000000000000000000,000
39 data 00000000000000000000000000,000
40 data 0000000000ad09ff8d09ffad0b,402
41 data ff186902a8cc1dffd0fba9198d,72c
42 data 15ffa9148d13ffa4d1cc1dffd0,79d
43 data fba9008d15ffa91c8d13ffe6d0,75f
44 data c6d1a5d08d0bffa5d0c966d003,81a
45 data 2042124cc3fc78a90ea0ce8d14,5bd
46 data 038c15035860000000000000000,15f
47 data 00000000000000000000000000,000
48 data 00000000000000000000000000,000
49 data 00000000000000000000000000,000
50 data 00000000000000000000000000,000
51 data 00000000000000000000000000,000
52 data 00000000000000000000000000,000
53 data 00000000000000000000000000,000
54 data 00000000000000000000000000,000
55 data 00000000000000000000000000,000
56 data 00000000000000000000000000,000
57 data 00000000000000000000000000,000
58 data 00000000000000000000000000,000
59 data 00000000000000000000000000,000
60 data 00000000000000000000000000,000
61 data 00000000000000000000000000,000
62 data 00000000000000000000000000,000
63 data 00000000000000000000000000,000
64 data 00000000000000000000000000,000
65 data 00000000000000000000000000,000
66 data 00000000000000000000000000,000
67 data 00000000000000000000000000,000
68 data 00000000000000000000000000,000
69 data 00000000000000000000000000,000
70 data 00000000000000000000000000,000
71 data 00000000000000000000000000,000
72 data 00000000000000000000000000,000
73 data 00000000000000000000000000,000
74 data 00000000000000000000000000,000
75 data 00000000000000000000000000,000
76 data 00000000000000000000000000,000
```

```
77 data 00000000000000000000000000,000
78 data 00000000000000000000000000,000
79 data 0000000000000000000020819d,13e
80 data a1a19d817e00003c66667e6666,530
81 data 00007c667c66667c00003e6060,3a4
82 data 60603e00007c363636367c0000,2ce
83 data 7e303c30307e00007e303c3030,312
84 data 3000003e60606e663e00006666,30c
85 data 7e666666000018181818181800,240
86 data 0006060666663c0000666c7878,2dc
87 data 6c660000060606060607e000066,396
88 data 7e7e666666000066767666e6e66,4c2
89 data 00003c666666663c00007c6666,358
90 data 7c606000003c66666e3c3e0000,32c
91 data 7c66667c6c6600003e603c0606,37c
92 data 7c00007e181818181800006666,23e
93 data 6666663e000066666663c3c1800,332
94 data 006666667e7e66000000663c183c,38a
95 data 666600006666 3c181818 00007e,29a
96 data 0e1830607e00003c303030303c,26c
97 data 00001c30307c307e00003c0c0c,1fa
98 data 0c0c3c0000183c7e1818180000,16e
99 data 10307e7e30100000000000000000,17c
100 data 0000000c0c1818003000006644,122
101 data 00000000000066ff6666ff6600,396
102 data 003e783c1e7c18000000264 0810,222
103 data 264000003078307e7c3600000c,27a
104 data 18000000000000183030303018,108
105 data 0000180c0c0c0c18000066187e,15c
106 data 18660000000018187e18180000000,15c
107 data 000000181830000000007e0000,0de
108 data 000000000000000181800000306,039
109 data 0c183060000003e666e76667c00,31e
110 data 00183838181818000003c66061c,194
111 data 307e00007c063c06067c000018,20c
112 data 30607e181800007e607c06067c,320
113 data 00003c607c66663c00007e660c,310
114 data 181818000003c663c66663c0000,22e
115 data 3c66663e063c00000018000018,1b8
116 data 00000000180000181800000c18,06c
117 data 30180c000000007e007e000000,150
118 data 0030180c183000000003c660c18,162
119 data 00180000ff000000000000c103c,16f
120 data 66667e66660000000000000000,216
121 data 00000000000000000000000000,000
122 data 0000000c107e303c30307e0000,1e4
123 data 00000000000000000000000000,000
124 data 006600666666666663e0c180018,2de
125 data 18181818336600666666663e66,335
126 data 003c6666666663c3366003o6666,3b1
127 data 663c0000000000000000000000,0a2
128 data 000000000000c103c66666663c,22c
129 data 00000000000000000000000000,000
130 data 00000000000000000000000000,000
131 data 00000000000000000000000000,000
132 data 000c1866666666663eeff0e1e3,603
133 data e3e1ffe7fe03fe30e0b0e08000,8c9
134 data 000000eff0e1e300000000fe03,4a4
135 data fe30e3e1ffe70000000000e0b0e0,748
136 data 80000000000000000000000000,080
137 data 00000000000000000000000000,000
138 data 00000000000000000000000000,000
139 data 0000000000007e7e3c3c3c3f7f,26e
140 data 7f0000000000efefefe7f7f3c3f,500
141 data 3f3c7f7ffefe0ef0f00efefe7f,7ec
142 data 7f7f7303030707fefefecec0c0,6cd
143 data e0e0071f3c3f3f3c7e7ee0f83c,5ec
144 data fcfc3c7e7e0000000000000000,330
145 data 7f7f3e3f3f3c7e7efcfe3efefc,724
146 data f0787e000000001010001000000,1e9
147 data c0e0e0e0c000000002130c283023,4bc
148 data 630040e41804056762 63f090b8,50c
149 data 8cd7c21e6502030 41ae8407800,46b
150 data 002828281830e000000a12141c,1ec
151 data 30e000002828281830e0010202,2b5
152 data 02020200000000000000000000,006
153 data 000000000000000015000071 00d,039
154 data 000000ff818199998181ff0000,534
155 data 3c3c3c3c0000007e7e66667e7e,3b4
156 data 00ffffc3c3c3c3ffff00000000,708
157 data 00000000000000000000000000,000
158 data 003c3c3c3c00001f2020202020,1af
159 data 20202020202020202020202020,1a0
160 data 20202020202020202020202020,1a0
161 data 20202020202020201f2020206e,1ed
162 data 6f7220200f120209142006 2020,1c7
163 data 6e6f72200021920020 90c0c0c19,1f2
164 data 2019011204132020201f202020,142
165 data 70712020204040404040404020,321
166 data 20707120202028032920140213,1fe
167 data 0f20313939302020202 01f4040,221
168 data 40404040404040404040404040,340
169 data 40404040404040404040404040,340
170 data 404040404040404040 40201f20,2df
171 data 3420020914051320 4b13131a05,13b
172 data 0e190f0d4f2010120f0712010d,10a
173 data 200c050805144c0c050720201f,115
174 data 20202020202020202020202020,1a0
175 data 20202020202020202020202020,1a0
176 data 20202020202020202020202020,1a0
177 data 1f202002410a140f1320050c16,129
178 data 050e204b13131a050e190f0d0f,115
179 data 1414200d4a16050b08051a2020,12c
180 data 201f20202020202020202020 20,19f
181 data 20202020202020202020202020,1a0
182 data 20202020202020202020202020,1a0
183 data 20201f56572006323a4b13131a,229
184 data 050e190f0d411320050c090e04,0e8
185 data 49144113012020202020202020,1b2
186 data 2020201f20202040402020 2020,1df
187 data 20202020202020202020202020,1a0
188 data 20202020202020202020202020,1a0
189 data 20202020 1f56572006313a1415,206
190 data 12024f13200d050e144513200b,14d
191 data 011a05141441120120202020 20,13c
192 data 2020202020 1f2020204040 2020,1df
193 data 20202020202020202020202020,1a0
194 data 20202020202020202020202020,1a0
195 data 202020202020 1f56572006333a,21f
196 data 0b05120513451320202020 2020,152
197 data 20202020202020202020202020,1a0
198 data 20202020202020201f20204040,1df
199 data 20202020202020202020202020,1a0
200 data 20202020202020202020202020,1a0
201 data 20202020202020201f56572008,1f4
202 data 050c103a0205144b0c14451320,159
203 data 20202020202020202020202020,1a0
204 data 20202020202020201f202020,19f
205 data 40404040202020202020202020,220
206 data 20202020202020202020202020,1a0
207 data 202020202020202020201f2020,19f
208 data 011a2005120504051409204513,0f5
209 data 20011a20550a200b051a044c2d,181
210 data 2045132016450 72d202020 1f20,1c6
211 data 20202020202020202020202020,1a0
212 data 20202020202020202020202020,1a0
213 data 202020202020202020202020 1f,19f
214 data 03490d050b05142001202 43202,11b
215 data 2d24320520024 10a140f0b0205,12a
216 data 0e200b05120513 13480b212020,12f
217 data 1f404040404040404040404040,31f
218 data 40404040404040404040404040,340
219 data 40404040404040404040404040,340
220 data 201f204804163a200b0f06410e,18a
221 data 010b20011a204b140c05144545,175
222 data 12142c20202020202020202020,192
223 data 20201f20202020202020202020,19f
224 data 20202020202020202020202020,1a0
225 data 20202020202020202020202020,1a0
226 data 2020201f2020202020200a0103,14d
227 data 0b200f270c010e1405120e0e05,0c8
228 data 0b2c200c051a0c090e050b2045,11a
229 data 132020201f2020202020202020,192
230 data 20202020202020202020202020,1a0
```

```
231 data 202020202020202020202020202020,1a0
232 data 20202020201f202020202020202001,180
233 data 071914124b131a140e050b200d,11d
234 data 0507200d0945121420 0e052e2e,13c
235 data 2e2020202020202030302030302020,1ee
236 data 303020303020303020303020302030,230
237 data 30203aae0000000000000000000000,138
238 data 000000000000000000000000000000,000
239 data 000000000000000000000000000000,000
240 data 000000000000000000000000000000,000
241 data 000000000000000000000000000000,000
242 data 000000000000000000000000000000,000
243 data 000000000000000000000000000000,000
244 data 000000000000000000000000000000,000
245 data 000000000000000000000000000000,000
246 data 000000000000000000000000000000,000
247 data 000000000000000000000000000000,000
248 data 000000000000000000000000000000,000
249 data 000000000000000000000000000000,000
250 data 000000000000000000000000000000,000
251 data 000000000000000000000000000000,000
252 data 000000000000000000000000000000,000
253 data 000000000000000000000000000000,000
254 data 000000000000000000000000000000,000
255 data 000000000000000000000000000000,000
256 data 000000000000000000000000000000,000
257 data 000000000000000000000000000000,000
258 data 000000000000000000000000000000,000
259 data 000000000000000000000000000000,000
260 data 000000000000000000000000000000,000
261 data 000000000000000000000000000000,000
262 data 000000000000000000000000000000,000
263 data 000000000000000000000000000000,000
264 data 000000000000000000000000000000,000
265 data 000000000000000000000000000000,000
266 data 000000000000000000000000000000,000
267 data 000000000000000000000000000000,000
268 data 000000000000000000000000000000,000
269 data 000000000000000000000000000000,000
270 data 000000000000000000000000000000,000
271 data 000000000000000000000000000000,000
272 data 000000000000000000000000000000,000
273 data 000000000000000000000000000000,000
274 data 000000000000000000000000000000,000
275 data 000000000000000000000000000000,000
276 data 000000000000000000000000000000,000
277 data 000000000000000000000000000000,000
278 data 000000000000000000000000000000,000
279 data 000000000000000000000000000000,000
280 data 000000000000000000000000000000,000
281 data 000000000000000000000000000000,000
282 data 000000000000000000000000000000,000
283 data 000000000000000000000000000000,000
284 data 000000000000000000000000000000,000
285 data 000000000000000000000000000000,000
286 data 000000000000000000000000000000,000
287 data 000000000000000000000000000000,000
288 data 000000000000000000000000000000,000
289 data 000000000000000000000000000000,000
290 data 000000000000000000000000000000,000
291 data 000000000000000000000000000000,000
292 data 000000000000000000000000000000,000
293 data 000000000000000000000000000000,000
294 data 000000000000000000000000000000,000
295 data 000000000000000000000000000000,000
296 data 000000000000000000000000000000,000
297 data 000000000000000000000000000000,000
298 data 000000000000000000000000000000,000
299 data 000000000000000000000000000000,000
300 data 000000000000000000000000000000,000
301 data 000000000000000000000000000000,000
302 data 000000000000000000000000000000,000
303 data 000000000000000000000000000000,000
304 data 000000000000000000000000000000,000
305 data 000000000000000000000000000000,000
306 data 000000000000000000000000000000,000
307 data 000000000000000000000000000000,000
308 data 000000000000000000000000000000,000
309 data 000000000000000000000000000000,000
310 data 000000000000000000000000000000,000
311 data 000000000000000000000000000000,000
312 data 000000000000000000000000000000,000
313 data 000000000000000000000000000000,000
314 data 000000000000000000000000000000,000
315 data 000000000000000000000000000000,000
316 data 00204fff931153544152543a00,3da
317 data a20020cfff9d000cc952f034c9,641
318 data 0df0e4e8e004d0ed20cfffc90d,82e
319 data d0f9a200a000b9000c208bfa95,60a
320 data 05c8b9000c208bfa4a4a4a4a15,474
321 data 059505e8c8c004d0e34c5406a9,615
322 data 8b8505a9dc8506a5058d730ba5,57f
323 data 068d720b788d3fffa2008a9d00,51c
324 data 08e8e030d0f8a52c85d1a90085,71d
325 data d0a42bb1d0290f20f10ab1d04a,63e
326 data 4a4a4a20f10ac8d002e6d1c42d,63b
327 data a5d1e52e90e3a2008a9def0be8,7a7
328 data e010d0f7a90f8540a5408541a4,683
329 data 40a641bdef0bbeef0ba8bd2008,623
330 data d92008f021b00bc64110e7c640,5d1
331 data 10df4cf306a640a441bdef0b48,5fe
332 data b9ef0b9def0b6899ef0b10e1bd,6f3
333 data 1008d91008f004b0e290d5bd00,5b1
334 data 08d90008f0cdb0d690c9a00084,6a9
335 data e684e784e84c480a0000ca10ef,624
336 data a90085d0a9f085d2a90c85d185,77e
337 data d3a204a000b1d291d0c8d0f9e6,874
338 data d1e6d3cad0f2a52d85d0a52e85,895
339 data d1a900a0fd85d284d320cf0ba5,764
340 data d0c52bd0f7a5d1c52cd0f1a910,868
341 data a00f85d084d1a00884d4a00084,67d
342 data d7a5d285d6a90085d2a4d6b1d2,8a6
343 data 4a4a4a4a200c0aa4d6b1d2290f,493
344 data 200c0ae6d6d0e8e6d3a5d3c9fd,8a1
345 data d0e0a4d7a5d4f00eaab1d00a91,868
346 data d0cad0f8c8d002e6d1981865d0,898
347 data 85d09002e6d1a5d01869ef85d2,7da
348 data 852d8de60ba5d1690185d3852e,61b
349 data 8de70b20cf0ba5d0c910d0f7a5,733
350 data d1c90fd0f1a900bdff0a9dff0f,784
351 data cad0f7a52ba42c8dea108ceb10,73f
352 data a901a010852b842c8d3effa910,53d
353 data 85d0a90f85d1a90085d2a91085,6a1
354 data d320cf0ba5d0c900d0f7a5d1c9,811
355 data 0cd0f18d3eff58208bd8a9418d,6e9
356 data 19ff4c0387a20fddef0bf004ca,634
357 data 10f8eabde00a853fa203a90006,5b1
358 data 3f2acad0faaaa4d7b1d0063f2a,712
359 data 91d08d19ffc6d4f004cad0ed60,87b
360 data a00884d4e6d7d0f4e6d1d0f0ea,9e2
361 data beef0b18a5e87d000885e8a5e7,6db
362 data 7d100885e7a5e67d200885e6c8,664
363 data c002d0e2a00cbeef0b38a5e8fd,79a
364 data 000885e8a5e7fd100885e7a5e6,70d
365 data fd200885e69024c8c010d0e0a2,72e
366 data 0346e666e766e8cad0f7a5e838,820
367 data e9fe85e8b008a5e7d004a5e6f0,8e7
368 data 034cf0088d3eff204fff931120,543
369 data 534f5252592e0d0d57414e5420,341
370 data 49543f00a9f72070dbc9fdf010,6ad
371 data a9ef2070dbc97fd0ee208bd858,7e4
372 data 4c03878d3fff4cf00860648888a,5bb
373 data acadaeaf9092949698 9a9c9eea,858
374 data aafe0008d008fe1008d003fe20,58f
375 data 0860000b1037099e3431303900,22f
376 data 0000a2e2bd1c109dff0bcad0f7,6a5
377 data ca9a4c000c788d3fffa208bdc9,62f
378 data 0c95d0ca10f820b20ca5d0c9ff,75e
379 data d0f7a5d1c910d0f1a5d285d7a9,953
380 data 0085d220570c0a0a0a0aa4d691,40d
381 data d420570ca4d611d491d4e6d6d0,7a7
382 data 02e6d5a5d3c9fdd0e08d3eff58,8cd
383 data a5d6852da5d5852e20be8b4cdc,6eb
384 data 8ba900853fa2032092 0ca53fc9,508
```

```
385 data 04b00dc903f01dc902f00eaabd,5ca
386 data d20c60a2012092 0ca53fd0f2a2,5e7
387 data 0120920ca53f4906d0e7a20220,46d
388 data 920ca53f2907d0dca4d7b1d20a,666
389 data 263f91d28d19ffc6d8f004cad0,799
390 data ed60a90885d8e6d7d0f4e6d3d0,965
391 data f0a5d0d002c6d1c6d0a5d2d002,8ad
392 data c6d3c6d2a000b1d091d2600000,715
393 data 00fd080c00000800010203040 5,128
394 data 060708090a0b0c0d0e0f00a90b,11d
395 data 8d3003a90d8d31036048a5aec9,4fb
396 data 03f004684c800fa52b8d7b0fa5,4c6
397 data 2c8d7c0fa52d8d7d0fa52e8d7e,50d
398 data 0fa5ad8dbf0ea00fa92099c00e,59a
399 data 8810faa4abd00aa90385b0a937,67c
400 data 85afa01020c80feaeaa048b977,6c7
401 data 0e99ff0688d0f72019e3a9488d,695
402 data 2403a9038d2503a901aaa820ba,45e
403 data ffa9bca2c0a00e20bdffa90085,77e
404 data 9aa9038523a9248522a922a003,4d0
405 data a22620d8ffa94b8d2403a9ec8d,689
406 data 2503a900aaa885b2a90785b385,5c7
407 data 9ea949859d20c80da980859aa2,691
408 data 00a000ad7b0f85b2ad7c0f85b3,57e
409 data ad7d0f859dad7e0f859e20c80d,5ad
410 data 4c038720120ea5b220400ea5b3,433
```

típusú packernek, ugyanis egy összenyomatlan programról bár többet szedne le, annak bájtjait úgy összezavarná, hogy egy bájtos packer már csak a karját tudná tőle sajnálkozóan széttárni. Ezért leghasznosabb a fenti módon kétszer összegyűrt programot adni az ORBIT F-nek.

Mielőtt azonban nekiesnénk, a biztonság kedvéért nyomjuk meg az F5-öt, s győződjünk meg arról, hogy a $2B—$2E címek a betöltött program kezdő- és végcímét tartalmazzák-e. Ha ez nem így lenne, írjuk át a $2B—$2E címeket. Ezután már nincs más teendőnk, mint kényelmesen hátradőlni és gyengéden megbillenteni az F2-t. A START után írjuk be a program kezdőcímét hezadecimálisan, dollárjel nélkül, avagy, ha az RUN-nal indul, a RUN szót. És íme: a program hosszától függően max. 1 percen belül megkapjuk az agyongyűrt programot. Ha READY felirat helyett a SORRY üzenetet kapjuk, hogy ez arra utal, hogy a program már agyon van gyűrve, s ha még az ORBIT F is meggyömködné, szegény csak megdagadna. Ha ezt vállaljuk, Y-t nyomjunk, ha viszont megesett a szívünk a programon és nem tesszük ki ennek a tortúrának, az N megnyomására érintetlenül visszakapjuk az eredeti programot. Ha pedig sikeres volt a tömörítés, érdemes ismét kipróbálni ugyanezen a programon (az így összenyomott program RUN-nal indul!), mert előfordult már, hogy egy programot többször is össze tudott tömöríteni az ORBIT F.

Ha készen vagyunk a tömörítéssel, jöhet a mentés. Lemezesek használhatják az F4-et, kazettásoknak az F1-et ajánlom; ez ugyanis turbóval menti az összenyomott programot kazettára. (Az ORBIT F ezért kazettás másolóprogramnak is kitűnően használható.)

És most arról, hogyan is működik a program. Tudjuk, hogy minden bájt 8 bitből áll, és hogy 8=4+4. Egy 4 bit, azaz nibble pedig 16 különféle értéket vehet fel. Az ORBIT F pontosan ezeket a nibble-ket kavargatja, innen a neve is. A 16 darab különböző nibble-t gyakorisági sorrendbe állítás után egyszerűen felcserélik egy-egy kóddal, mégpedig a leggyakoribb 4 bit 3 bites kódot kap, míg a legritkábbak 5 bitest. Ez kedvező nibble-eloszlás esetén kevesebb bitet eredményez, mint amennyi eredetileg volt. Kedvezőtlen eloszlás esetén pedig... Ám a programok nagy többsége elviseli ezt a tömörítőt legalább egyszer.

Kívánok tehát sikeres használatot a programhoz:

Nagy Attila

```
411 data 20400ea59d20400ea59e20400e,3cf
412 data 84b4a4b2a90085b2b1b220400e,63f
413 data c8d002e6b3c49da5b3e59e90ee,8ed
414 data a5b420400e20610ea91b8d06ff,4ac
415 data 8d3effa9088501582084ff6078,5d4
416 data 8d3fffa9008501a90b8d06ffca,60a
417 data d0fd88d0faa9a08d02ffa9008d,82c
418 data 03ffa9108d09ffa0ffa9102040,608
419 data 0e88d0f8a95a85b645b485b420,6ee
420 data 6f0eea06b6a50109022061 0ea2,405
421 data 2029fd20610ea21dc6b7d0ea60,62b
422 data cad0fd9006a212cad0fdea8501,7e8
423 data 60a212a90885b7600000a9f68d,58d
424 data fcffa9ff8dfdff8d3fffee19ff,9fd
425 data 18204a038d3effa9ee8d19ff58,5e3
426 data 208aff2084ffa59ec59dd019ad,787
427 data 4707f00ea9ff8d0cff8d0dff20,645
428 data be8b4cdc8b209d8a4c0387a21d,5d8
429 data 4c8386001b4e5351554545 5a45,3e0
430 data 52204a522e2020203878082093,307
431 data 0320b803a8852ba90085b220b8,4ee
432 data 03852cea85b320b803852d20b8,53b
433 data 03852e20b80391b2459e859ec8,5a2
434 data d002e6b38d19ffc42da5b3e52e,76c
435 data 90e720b803859da908850128b0,583
436 data 01604c0007a9008501cad0fd88,502
437 data d0fa20e603ea20c50326aca5ac,6c8
438 data c910d0f520b803c910f0f9c95a,75e
439 data d0ea60a20820c50326b6cad0f8,71a
440 data a5b660a9102401f0fc2401d0fc,676
441 data 2d09ff48a9e08d02ffa9008d03,5cd
442 data ffa9108d09ff6869fe60849e8c,72a
443 data 06ff60000000000000000040010,179
444 data 00a5aec901f005a5ae4ca4f120,666
445 data 19e3b029202 8f2a203a5ad2901,530
446 data d002a20186f8206ce5b015a900,5d2
447 data 85f820ae0f4c54f22019e32064,58c
448 data e3208de3b007a90085f74cbfe5,73f
449 data 4ceae5eaeaeaea60c0119002a0,826
450 data 108830f6788d3fffb1af99c00e,6c8
451 data 8d3eff584cce0f4c000deaeaea,662
452 data eaeaeaeaeaeaa210bd3603293f,78c
453 data 9dff0bcad0f5f0e500000000a2,6ad
454 data 03bdf30b48bdfb0b9df30b689d,669
455 data fb0b204fff53c122222c339191,54d
456 data 004ced0f000000000000000000,148
457 data 00000000000000000000000000,000
458 data 00000000000000000000000000,000
459 data 00000000000000000000000000,000
460 data 00000000000000000000000000,000
461 data 00000008080806070405035359,0dd
462 data 5333331320d4dcf0d47363030,32f
463 data 0d5359533130323 40d44534156,30e
464 data 45224dcf0d3e32420d52554e0d,351
465 data 4c4953540d4ccf0d0000000000,271
466 data 00000000000000000000000000,000
467 data 00000000000000000000000000,000
468 data 00000000000000000000000000,000
469 data 00000000000000000000000000,000
470 data 00000000000000000000000000,000
471 data 00000000ff00ff00ff00ff00ff,4fb
472 data 00000c0000ff00ff00ffa205a9,459
473 data 808df105a9038df0201be3208d,5f7
474 data e3a90b8d06ffa501293f4a4a4a,515
475 data 4a09408d19ffad4305f0eea91b,5cf
476 data 8d06ff20b0e36000000000000,3a5
63000 scnclr:s=4096:k=0:fori=1to476:l=0
63001 reada$:readb$
63002 forj=0to12:b=dec(mid$(a$,2*j+1,2))
63003 pokes+k,b
63004 k=k+1:l=l+b
63005 next:print"";476-i" "
63006 if l<>dec(b$)thenprint"hiba!!! -->"i".
sor!!":stop
63007 next
63008 print"keszen vagyok!"
63009 poke43,1:poke44,16:poke45,38:
poke46,40:list
```

Lábjegyzetek az MPS-1230 Felhasználói kézikönyvéhez

Mindenekelőtt — pusztán emlékeztetőül — idézzük fel mindazt, amit az ASCII kódokról tudni érdemes.

Az American Standard Codes for Information Interchange (az információcsere kódjainak amereikai szabványa) 1963-ban került bevezetésre. A 0 és 127 közötti intervallumban, 128 karakter helyét definiálja.

1. TÁBLÁZAT.

00	NUL	karakter
01	SOH	fejléc kezdete
02	STX	szöveg kezdete
03	ETX	szöveg vége
04	EOT	átvitel vége
05	ENQ	állomás hívása
06	ACK	nyugtázás
07	BEL	hangjelzés
08	BS	visszaléptetés
09	HT	vízszintes tabulálás
10	LF	soremelés
11	VT	függőleges tabulálás
12	FF	lapemelés
13	CR	kocsivissza
14	SO	váltó ki
15	SI	váltó be
16	DLE	adatátvitel átkapcsolása
17	DC1	1. készülékvezérlő
18	DC2	2. készülékvezérlő
19	DC3	3. készülékvezérlő
20	DC4	4. készülékvezérlő
21	NAK	negatív nyugtázás
22	SYN	szinkronizálás
23	ETB	átviteli blokk vége
24	CAN	érvénytelenítés
25	EM	hordozóeszköz vége
26	SUB	helyettesítés
27	ESC	escape kód
28	FS	csoportelválasztó
29	GS	csoportelválasztó
30	RS	rekordelválasztó
31	US	egységelválasztó
127	DEL	törlés (delete)

Az 1. táblázat 33 (0—31. valamint a 127.), úgynevezett vezérlőkaraktert sorol fel. A karakterek többsége csak a professzionális adatforgalomban (pl. távközlés) használatos, és csupán a kisebbik hányada a — minket is érdeklő — nyomtatóvezérlő kód.

A 2. táblázat a vezérlőkódok szimbólumait is tartalmazza. Közülük csupán a 3—6. kódú (kör; káró; treff; pikk) nyomtatható, de csakis az IBM karakterkészlettel dolgozó nyomtatókkal. Ugyanakkor az IBM PC-család képernyőin megjeleníthetőek.

2. TÁBLÁZAT

A 2. táblázat már átvezet az ún. bővített ASCII kódtábla fogalmához. Mint látható, ez a táblázat további 128 karaktert tartalmaz, a 128—255 közötti intervallumban. Megjelennek a „külföldiek'' által használt karakterek, így néhány magyar írásban használatos ékezetes karakter, továbbá görög betűk, és tetszetős keretező karakterek.

Az MPS-1230 printer, csak a PROPRINTER és az IBM.G.P. emulációban tudja elérni az IBM. INTERNATIONAL 1. karakterkészletet. Márpedig ez a karakterkészlet csak 2 karakterrel (249 és 250) tér el az eredeti bővített ASCII karaktertáblától. Azonban, eléggé el nem ítélhető módon az ASCII táblázat — a nevében deklarált szabvány jellege ellenére is — csupán ajánlás.

A legnagyobb eltérést az EPSON emuláció, valamint a COMMODORE-command által elérhető kódtáblákban tapasztalhatjuk. Lényeges különbség például a nagy-, és kisbetűk felcserélése; a 0—31. közötti vezérlőkarakterek ismétlése a 128—143 közötti intervallumban; a keretező karakterek hiánya.

Ráadásul van még egy alattomos — de szerencsére kivédhető — eltérés is. Nevezetesen a HC gépek, minden logikai sor kiíratása után kiadnak egy chr$(13) kódot. (Kocsivissza + soremelés.) Csakhogy ezt a kódot minden PC. „kocsivisszá''-nak értelmez, soremelés nélkül. Így például egy program listája egyetlen, fekete sáv lesz. Legegyszerűbben úgy egalizálhatjuk ezt az eltérést, ha a SET UP eljárás során az alábbi ajánlatokat fogadjuk el:

LF = LF + CR; CR = CR + LF

A továbbiakban egy érdekességre szeretném felhívni a figyelmet. Az alábbi sor egy részlet az IBM INTERNATIONAL 1 kódtáblából.

Ç é á ░ └ ╨ α ≡

Figyeljük meg a balról negyedik, keretező karaktert ⟨chr$(176)⟩. Tessék utánaszámolni: 11 sorból áll!! Persze ez szükségszerű, hiszen egy függőleges keretvonalnak a sorközöket is le kell fednie. (Következésképpen több tizenegysoros karakter van még.)

Próbáljuk meg „leleplezni a trükköt''! Mindenekelőtt a SET UP eljárással jelöljük ki az alábbiakat:

SERIAL COMMODORE
PROPRINTER vagy IBM G.P.
INTERNATIONAL 1
USA/UK,

valamint a fentebb említett LF/CR variációkat.

Ezután írjuk be az alábbi pár soros programocskát:

```
10 open 1,4
20 print#1," *"chr$(176);tab(70)"*"
30 close1
```

Futtatva a programot, kinyomtatódik a két csillag, közöttük a tizenegysoros karakterrel. A nyomtatófej a baloldalon marad. Most pedig koncentráljunk! Tegyük ujjunkat a nyomtató LOCAL gombjára, és indítsuk el ismét a programot. Amint kiíródott a keretező karakter, azonnal nyomjuk meg a LOCAL-t. A jobboldali csillag még kinyomtatódik, de a nyomtatófej visszatértében félúton megáll. Ekkor megfigyelhetjük, hogy a keretező karakter még normál magasságú. Újra megnyomva a LOCAL-t, a nyomtatófej pontosan a helyére nyomtatja a karakter hiányzó, sorköznyi darabját.

Befejezésül, még egy érdekesség. Írjuk be az alábbi programot:

```
10 rem P.C. keret
20 :
30 open4,4:open5,5,7:goto 70
40 print#4,"  "chr$(186)tab(35)chr$(186);:return
50 for l=1 to 35
60 print#4,chr$(205);:next l:return
70 print#4,"  "chr$(201);:gosub 50
80 print#4,chr$(187)
90 for k=1 to 4:gosub 40
```

```
100 n=0:if k/2=int(k/2) then n=32
110 print#5,tab(9);
120 for l=65 to 90:print#5,chr$(l + n);:next l
130 print#4:next k:gosub 40
140 print#4,chr$(10)" "chr$(200);
150 gosub 50:print#4,chr$(188)
160 close4:close5:end
```

Ha a fentebb javallott SET UP eljárás után lefuttatjuk a programot, az alábbi printeroutputot kapjuk:

```
abcdefghijklmnopqrstuvwxyz
ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrstuvwxyz
ABCDEFGHIJKLMNOPQRSTUVWXYZ
```

Persze első blickre mindez öncélú, gyermekded játéknak tűnhet. Ám ha a teljes magyar ékezetes karakterek mellett, a P.C. keretező karaktereire is igényt tartunk, máris értelmet nyert a fenti rutin. De erről talán egy más alkalommal…

Szász Sándor

		0	16	32	48	64	80	96	112	128	144	160	176	192	208	224	240
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	0		►	Space	0	@	P	‘	p	Ç	É	á	░	└	╨	∝	≡
1	1	☺	◄	!	1	A	Q	a	q	ü	æ	í	▒	┴	╤	β	±
2	2	☻	↕	"	2	B	R	b	r	é	Æ	ó	▓	┬	╥	Γ	≥
3	3	♥	‼	#	3	C	S	c	s	â	ô	ú	│	├	╙	π	≤
4	4	♦	¶	$	4	D	T	d	t	ä	ö	ñ	┤	─	╘	Σ	⌠
5	5	♣	§	%	5	E	U	e	u	à	ò	Ñ	╡	┼	╒	σ	⌡
6	6	♠	▬	&	6	F	V	f	v	å	û	ª	╢	╞	╓	µ	÷
7	7	•	↨	’	7	G	W	g	w	ç	ù	º	╖	╟	╫	τ	≈
8	8	◘	↑	(	8	H	X	h	x	ê	ÿ	¿	╕	╚	╪	Φ	°
9	9	○	↓	)	9	I	Y	i	y	ë	Ö	⌐	╣	╔	┘	Θ	∙
10	A	◙	→	*	:	J	Z	j	z	è	Ü	¬	║	╩	┌	Ω	·
11	B	♂	←	+	;	K	[	k	{	ï	¢	½	╗	╦	█	δ	√
12	C	♀	∟	,	<	L	\	l	|	î	£	¼	╝	╠	▄	∞	ⁿ
13	D	♪	↔	-	=	M	]	m	}	ì	¥	¡	╜	═	▌	φ	²
14	E	♫	▲	.	>	N	^	n	~	Ä	₧	«	╛	╬	▐	∈	■
15	F	☼	▼	/	?	O	_	o	⌂	Å	ƒ	»	┐	╧	▀	∩	

Az eredeti amerikai ASCII kódtáblázat

*Az 1. táblázat az Oxford Számítástechnikai Értelmező Szótár (NOVOTRADE KIADÓ 1989.) ASCII szócikke alapján készült. A 2. táblázat az ALAPLAP 1990/6. számának 4. oldaláról származik.

Hexadecimális táblázat készítés a memóriában

91/11. szám 13. oldalán Egy trükk címmel azt fejtegettem, hogy ezt a speciális feladatot a READ-DATA kombináció helyett kevesebb írással meg lehet oldani. A javasolt számátalakítás azonban lassú. A következtetésem azt volt, hogy a kettő közül kell választani. Ez sem igaz!

Van egy harmadik lehetőség is: a szövegátalakítással! Ez is rövid rutin, és kb. olyan gyors, mint a RAD-DATA mód. Csodálkozással említem meg, hogy a 2. és 3. változattal a szakirodalomban még nem találkoztam! Talán a szerzőket a programhossz alapján díjazzák. Egyébként elismerem, a DATA sorok a sok idézőjellel és vesszővel nagyon jól mutatnak; és becsülettel töltik a lemezek blokkjait.

Teljesség kedvéért mindhárom megoldást leírom.

Stopper nélkül is érezhető a különbség! Van más mód is? Mivel a saját hibák bölcsebbé teszik az embert, azt kell mondanom, nincs kizárva!

Íme egy (talán) ideillő vicc!

Egy gyereknek azt mondja az apja: te, fiam, hol jártál, hogy térdig sáros vagy; hiszen ebben a kánikulában nincs is sár!

A fiú válasza: dehogy nincs apám, csak keresni kell!

A BASIC nyelv is ,,aszályosnak'' tűnik, mégis érdemes keresni…

Mesterházi Sándor

```
1. változat:
          10 DIM H$(255):REM READ-DATA MEGOLDAS
          100 FORI=0TO255:READH$(I):NEXTI
          30000 DATA "00","01","02",....
          .......
          30090 DATA ........,"FD","FE","FF"
2. változat:
          1159 DIM H$(255):REM '91/11. SZAMATALAKITASSAL
          1160 FORI=0TO255:H=I
          1161 FORJ=1TO0STEP-1
          1162 A=INT(H/(16↑J)):H=H-(A*(16↑J))
          1163 H$(I)=H$(I)+CHR$(A+48-(A>9)*7)
          1164 NEXTJ:PRINT I" = "H$(I)
          1165 NEXTI
3. változat:
          1159 DIM H$(255):REM SZOVEGATALAKITASSAL
          1160 A$="0123456789ABCDEF":O=0
          1161 FORI=0TO240STEP16:O=O+1
          1162 FORJ=0TO15
          1163 H$(I+J)=MID$(A$,O,1)+MID$(A$,J+1,1)
          1164 PRINTI+J" = "H$(I+J):NEXTJ
          1165 NEXTI
```

Directory kezelő

Ez a kis ,,utility program'' hasznos szolgáltatásokat tehet, ha a lemez BAM-jébe szeretnénk ,,belenyúlni''. Működése legegyszerűbben az egyes funkciókon keresztül mutatható be.

1. Directory. Kiírja a file-ok neveit, típusait, ,,('-jellel az esetleges törlés elleni védelmet.
   Egy menetben max. 20 file nevet ír ki. A kiírás (F—1)-re folytatódik.
2. Lemez átnevezés.
3. File átnevezés.
   Mindkét átnevezéskor a névképzés általános szabályait kell követni. (Pl. LEN (,,NÉV'')(=16)
4. File védelem be.
5. File védelem ki.
6. File törlés. Mindig megerősítést vár a törlési szándékra vonatkozóan. Igenlő válasz esetén, védett file-ot is töröl.

A file átnevezés és törlés a megszokott BASIC utasításokkal történik, míg a többi funkció az úgynevezett közvetlen elérésű utasításokat használja. Ezeket az utasításokat nagyon pontosan kell beírni a programba, különben nagyon csúnyán ,,belepiszkíthatunk'' a BAM-be. Netán olyannyira, hogy a lemez használhatatlanná válik. (Persze nem fizikailag.) Célszerű tehát, szolgálaton kívüli lemezekkel végezni a programbelövést.

A műveletek kijelölése után, a file nevét kéri a program. Elegendő az első néhány karaktert megadni. Ha a program a megadott karakterekkel kezdődő nevet talál, a teljes file nevet ,,kínálja'' fel. Ha nem fogadjuk el, tovább keres. A végrehajtás akkor kezdődik el, ha egy névre igennel válaszolunk.

A program kokkrekt működéséhez ,,tisztába tett'' tartalomjegyzék szükségeltetik. Emiatt minden lemezen először egy VALIDATE processzus zajlik le. (A file-ok újraszervezése.)

FIGYELEM! Ha a lemezen véletlen-file-ok vannak, azok blokkjai törlődnek!!

Szász Sándor

Felhasznált irodalom: A VC-1541-es lemezegység programozása DATA BECKER — NOVOTRADE Kiadó.

```
0 rem dir kezelo/920317
10 :
20 poke 53280,0:poke 53281,0:poke 808,234
30 print chr$(158)chr$(14)chr$(8)
40 pu$="                "
50 ty$(1)="SEQ":ty$(2)="PRG":ty$(3)="USR":ty$(4)="REL"
60 me$(1)="Directory           [1]":me$(2)="Lemez atnevezes   [2]"
70 me$(3)="File atnevezes      [3]":me$(4)="File vedelem be   [4]"
80 me$(5)="File vedelem ki     [5]":me$(6)="File torles       [6]"
90 me$(7)="E N D !             [7]":me$(8)="Valasszon           [ ]":goto 200
100 open2,8,2,"#"
110 print#15,"u1 2 0";tt;ss
120 print#15,"b-p 2";p:return
130 print " █I█/█N"
140 get w$:if w$<>"i" and w$<>"n" then 140
150 return
160 poke 19,1:input f$:poke 19,0:print
170 ho=len(f$):if ho<=16 then 190
180 f$=left$(f$+pu$,16)
190 return
200 print"███Dolgozom !!"
210 open15,8,15,"i"
220 input#15,e,e$:if e=0 then 240
230 print "█ "e$" ":goto 960
240 tt=18:re=1:ss=0:gosub 100:input#15,x1$
250 if val(x1$)=26 then 960
260 p=144:gosub 110
270 ln$="":for l=1 to 16:get#2,a$
280 if a$=chr$(160) then l=16:goto 300
290 ln$=ln$+a$
300 next l:if ln$=un$ then 330
310 un$=ln$:close2:print#15,"v"
320 goto 240
330 print "███:for l=1 to 8
340 print spc(9)me$(l)"█:next l
350 wait 198,1:get w$
360 q=val(w$):if q<1 or q>7 then 310
370 print"███:if q=7 then 960
380 if q>2 then 430
390 print "A lemez neve: █ "ln$" █:if q=1 then f1=1:goto 450
```

```
400 print "■■■ Az uj lemeznev: ■ ";:gosub 160:un$=f$
410 w$=left$(f$+pu$,16):tt=18:ss=0
420 p=144:gosub 120:print#2,w$;:goto 920
430 print"■ A file neve: ■ ";
440 gosub 160:qf=0:fl=0
450 t=18:tt=t:s=1:so=3:ss=s
460 p=0:gosub 110
470 get#2,x$:if x$="" then x$=chr$(0)
480 t=asc(x$):if t<>0 then tt=t
490 get#2,x$:if x$="" then x$=chr$(0)
500 s=asc(x$):if s<>255 then ss=s
510 for x=0 to 7
520 p=x*32+2:gosub 120
530 get#2,x$:if x$="" then x$=chr$(0)
540 if asc(x$)=0 then 760
550 z=asc(x$)
560 p=x*32+5:gosub 120
570 ff$="":for y=0 to 15
580 get#2,x$:if x$="" then x$=chr$(0)
590 if asc(x$)=160 then 610
600 ff$=ff$+x$
610 next y:if fl=0 then 690
620 so=so+1:print left$(ff$+pu$,16);
630 ve=0:if (z and 64)=0 then 650
640 z=z-64:ve=1
650 print " "ty$(z-128);:if ve then print" <":goto 670
660 print
670 if so/18<>int(so/18) then 760
680 so=0:gosub 970:print"■■■:goto 760
690 if f$<>left$(ff$,ho) then 760
700 print"■■"f$
710 print"■■O.K. a file? ";:gosub 130
720 f=8:if w$="n" then f=6:goto 740
730 qf=x+1:x=7
740 for l=f to 15:poke 781,l:sys 59903:next l
750 poke 214,f:sys 58640
760 next x:if qf then 790
770 if t<>0 then 460
780 if fl=0 then print"■ A file nem letezik !! ":goto 960
790 if q=1 then 960
800 if q>3 then p=(qf-1)*32+2:goto 830
810 print"■ Az uj filenev: ■ ";:gosub 160
820 print#15,"r:"+f$+"="+ff$:goto 950
830 if (z and 64)=0 then ve=0:goto 870
840 ve=1:if q=5 then 900
850 print"■■A file vedett !":if q=4 then 960
860 goto 880
870 if q=4 then k=z+64:goto 910
880 if q=6 then print"■■■ Toroljem ? ■";:gosub 130:if w$="n" then 960
890 if ve=0 then 940
900 k=z-64
910 gosub 120:print#2,chr$(k);
920 p=0:gosub 120
930 print#15,"u2 2 0";tt;ss
940 if q=6 then print#15,"s:"+f$
950 print "■■left$(me$(q),16)"vegrehajtva !!"
960 re=0:close2:close15:if q=7 then end
970 print spc(17)"■<F-1>
980 get w$:if w$<>chr$(133) then 980
990 if re then return
1000 q=0:goto 200

ready.
```

80 KARAKTERES JEGYZETTÖMB

```
1 rem *******************************
2 rem"* C= Ujsag              Ssz.:    *
3 rem"*    Jegyzettomb 80    C-128    *
4 rem"* Program:          Szolnoki Bela *
5 rem *******************************
6 :
7 :
8 print chr$(147)"data beolvasas es ellenorzes ...";:j=49152:ve=49976:p=j
9 for b=0 to 7:read a$
10 l=asc(mid$(a$,2,1))
11 h=asc(mid$(a$,1,1))
12 l=l-48:if l>9 then l=l-7
13 h=h-48:if h>9 then h=h-7
14 print"[illegible]"p;:p=p+1
15 if h>15 or l>15 then 17
16 a=h*16+l:poke j+b,a:t=t+a:next b:read a:if a=t then 18
17 print:print"data hiba ...  sor:"peek(64)*256+peek(63):end
18 t=0:j=j+8:if j<ve then 9
19 print"[illegible]kesz":end
20 data a9,18,85,e4,a9,00,85,e5,1085
21 data 85,e6,a9,4f,85,e7,a9,06,1150
22 data a2,22,20,cc,cd,a9,58,a2,1056
23 data 23,20,cc,cd,a2,00,bd,6b,0934
24 data 04,20,d2,ff,e8,e0,5b,d0,1256
25 data f5,a9,02,85,e5,a9,93,20,1126
26 data d2,ff,a9,01,a2,1a,20,cc,1059
27 data cd,ad,c6,04,c9,00,d0,12,1007
28 data a9,60,8d,a6,05,20,c7,04,0812
29 data a9,4c,8d,a6,05,a9,ff,8d,1122
30 data c6,04,ea,ea,ea,a5,91,c9,1415
31 data 7f,f0,03,4c,40,fa,a9,c6,1127
32 data 8d,02,03,a9,4d,8d,03,03,0539
33 data 4c,40,fa,1b,52,0e,1b,45,0609
34 data 1b,55,93,90,12,20,20,20,0517
35 data 20,20,20,20,20,20,20,20,0256
36 data 20,20,20,20,20,20,20,20,0256
37 data c3,4f,4d,4d,4f,44,4f,52,0736
38 data 45,20,c3,31,32,38,20,2d,0528
39 data 20,38,30,20,2d,20,c3,48,0512
40 data 41,52,41,43,54,45,52,20,0546
41 data 2d,20,c4,49,53,50,4c,41,0650
42 data 59,20,20,20,20,20,20,20,0313
43 data 20,20,20,20,20,20,20,20,0256
44 data 20,20,20,20,20,92,00,a9,0475
45 data 00,85,fb,85,fd,a9,13,85,1091
46 data fc,a9,19,85,fe,a2,03,a0,1158
47 data 00,b1,fb,91,fd,88,d0,f9,1419
48 data e6,fc,e6,fe,ca,d0,f2,a9,1787
49 data 13,85,fc,a9,16,85,fe,a2,1144
50 data 03,b1,fd,91,fb,88,d0,f9,1422
51 data e6,fc,e6,fe,ca,d0,f2,a9,1787
52 data 19,85,fc,a9,16,85,fe,a2,1150
53 data 03,b1,fb,91,fd,88,d0,f9,1422
54 data e6,fc,e6,fe,ca,d0,f2,a9,1787
55 data 24,a2,07,20,cc,cd,a9,10,0831
56 data a2,14,20,cc,cd,8d,2f,0a,0821
57 data a9,21,a2,06,20,cc,cd,a9,0980
58 data 32,a2,12,20,cc,cd,a9,03,0843
59 data a2,13,20,cc,cd,a9,10,20,0839
60 data ca,cd,a9,22,a2,12,20,cc,1026
61 data cd,a9,03,a2,13,20,cc,cd,0999
62 data a9,10,20,ca,cd,a9,10,a2,0971
63 data 12,20,cc,cd,a9,00,a2,13,0809
64 data 20,cc,cd,85,fb,a0,0b,a9,1165
65 data 80,20,ca,cd,c6,fb,d0,f9.1473
66 data 88,d0,f6,a9,07,a2,12,20,0978
67 data cc,cd,a9,d0,a2,13,20,cc,1203
```

Amikor az ember programoz, gyakran hiányzik egy olyan jegyzetblokk, amelyre rövid emlékeztetőket lehet felírni. A Jegyzettömb 80 a képernyőt nyolc sorral nagyobbítja meg. Ebbe a nyolc sorba rakhatjuk le a jegyzeteinket. Két oldal áll a rendelkezésre, amelyeket a ⟨SHIFT⟩, a ⟨CTRL⟩ és a ⟨CBM⟩ billentyűk egyszerre történő megnyomásával lehet fölülírni. Ha megnyomjuk ezeket a gombokat, akkor a képernyőn a 17—24. sorban álló információkat a látható jegyzetblokkba másoljuk. Az oldalak között a ⟨SHIFT⟩ ⟨CTRL⟩-lel lehet váltani. A ⟨CTRL⟩ ⟨CBM⟩ megnyomásával a látható jegyzetblokk sorai a 17—24. sorokba íródnak, és azokat ott editálni lehet. Azaz ettől kezdve semmi akadálya annak, hogy a 64-es BASIC programokat a 128-as módban, annak minden editálási komfortjával egyetemben adjuk be.

A jegyzettömb két oldala a \$1300—\$1580 és a \$1600—\$1880 területeken állnak. Az alábbi BASIC parancsokkal ezeket rendkívül egyszerűen lehet lemezre menteni:

BANK 0:BSAVE"oldal1",P(DEC("1300")) TO P(DEC("1580"))
BANK 0:BSAVE"oldal2", P(DEC("1600")) TO P(DEC("1880"))

Az oldalak betöltése egyszerűen a BLOAD "oldalX" paranccsal lehetséges. Akinek nem tetszenek a képernyő színek, az a POKE 49203, szín utasítással a keretet és a karaktereket átszínezheti. A POKE 49499,x paranccsal a felső szélt lehet beállítani. Az x értéke kb. 35 körül legyen, de az nagyban függ a csatlakoztatott

monitortól. A fenti három POKE utasítás hatása csak a SYS 49884 után lép életbe.

A program a tárolóban a 49152—50176 területen áll. Az indítás után bemásolja magát a (nem használt) 40 karakteres képernyőbe az 1024-es címtől kezdve. Ezt a területet tehát nem szabad fölülírni. Ha a Jegyzettömb 80 programot a SYS 49884 utasítással inicializáltuk, a 49152—50176 között álló területet nem szabad használni.

Sok örömet kívánunk ehhez a sok segítséget kínáló bővítéshez!

```
68 data cd,a9,00,85,fb,86,fc,a9,1313
69 data 02,85,fd,b1,fb,20,ca,cd,1255
70 data c8,d0,f8,e6,fc,c6,fd,d0,1797
71 data f2,a0,00,b9,00,15,20,ca,0842
72 data cd,c8,c0,80,d0,f5,4c,65,1355
73 data fa,a9,05,a2,12,20,cc,cd,1045
74 data a9,00,a2,13,20,cc,cd,85,0924
75 data fb,a8,a9,02,85,fd,86,fc,1362
76 data 20,d8,cd,91,fb,c8,d0,f8,1505
77 data e6,fc,c6,fd,d0,f2,a9,15,1573
78 data 85,fc,20,d8,cd,91,fb,c8,1434
79 data c0,80,d0,f6,4c,65,fa,a9,1370
80 data 05,a2,12,20,cc,cd,a9,00,0795
81 data a2,13,20,cc,cd,85,fb,86,1140
82 data fc,a9,02,85,fd,a0,00,b1,1146
83 data fb,20,ca,cd,c8,d0,f8,e6,1576
84 data fc,c6,fd,d0,f2,a9,15,85,1476
85 data fc,b1,fb,20,ca,cd,c8,c0,1511
86 data 80,d0,f6,4c,65,fa,a9,01,1179
87 data 85,e5,38,a5,61,e9,02,85,1048
88 data 61,b0,02,c6,62,a9,3f,8d,0944
89 data 00,ff,a9,00,a8,aa,85,fe,1149
90 data b1,61,f0,28,c9,20,d0,03,0998
91 data c8,d0,f5,85,ff,8a,29,07,1227
92 data aa,f0,0e,48,a5,ff,18,2a,0982
93 data 69,00,ca,d0,f9,85,ff,68,1256
94 data aa,e8,a5,ff,18,65,fe,85,1334
95 data fe,18,90,dc,c0,02,30,db,1103
96 data a9,00,8d,00,ff,c6,eb,a5,1163
97 data eb,48,20,7d,ff,12,13,3c,0816
98 data 00,a6,fe,a9,00,20,32,8e,0813
99 data 20,7d,ff,3e,1b,51,0d,00,0595
100 data 68,85,eb,a9,8d,20,d2,ff,1279
101 data 18,a5,61,69,02,85,61,90,0767
102 data 02,e6,62,a9,02,85,e5,4c,0939
103 data c6,4d,a5,d3,c9,07,d0,03,1070
104 data 4c,a9,05,c9,06,d0,03,4c,0744
105 data df,05,c9,05,d0,03,4c,c7,0920
106 data 04,c9,08,d0,03,4c,c2,06,0700
107 data c9,09,d0,03,4c,cf,06,4c,0786
108 data 65,fa,a9,c6,a2,4d,8d,02,1100
109 data 03,8e,03,03,4c,65,fa,a9,0747
110 data 16,a2,06,8d,02,03,8e,03,0481
111 data 03,4c,65,fa,a9,00,85,fb,0983
112 data 85,fd,a9,c0,85,fc,a9,04,1305
113 data 85,fe,a2,04,a0,00,b1,fb,1141
114 data 91,fd,88,d0,f9,e6,fc,e6,1703
115 data fe,ca,d0,f2,4c,ff,06,a9,1412
116 data 00,8d,00,ff,78,a2,04,8d,0823
117 data 18,03,8e,19,03,a9,9a,a2,0682
118 data 06,8d,14,03,8e,15,03,a9,0505
119 data 16,a2,06,8d,02,03,8e,03,0481
120 data 03,58,a9,00,8d,c6,04,a9,0772
121 data 60,8d,68,04,8d,5b,04,20,0613
122 data 00,04,a9,4c,8d,68,04,8d,0639
123 data 5b,04,60,ff,ff,ff,00,00,0956
READY.
```

Hosszú vezeték okozta zavarok

Gyakran használnak dugaszolókból és lapos kábelből álló csatlakozó hosszabításokat abból a célból, hogy a számítógép mögötti helyet szabadon hagyják, és az egyes perifériákat kissé távolabb helyezhessék el. Olyankor is szükség van erre, amikor a C64-et PC-házba helyezik el.

A lapos kábel egyes ereinek erős egymásra hatása miatt üzem közben az 1764, 1750 memória bővítők működésében zavarok, hibák lépnek fel.

Ezeket tehát közvetlenül a modulkapuknál használjuk bővítőn (expanderen) keresztül.

Helytelen kábel: lassú lemezegység

A közismerten gyors 1571 és 1581 lemezegységek és C128 együttes üzeménél gyakran állapíthatjuk meg, hogy ezek sajátos módon a C64-nél szokásos lassú üzemmódban dolgoznak.

Ennek az az oka, hogy az összekötő kábelt elcserélték. Az 1541-hez és a legtöbb nyomtatóhoz csak egy ötpólusú kábelt mellékelnek, amikben nincs bekötve az SRQ vezeték. Mivel azonban gyors üzemmódban erre a bekötésre is szükség lenne, a lemezegység átkapcsol lassú (C64—1541) üzemmódba.

A megoldás rendkívül egyszerű: kapcsolja a lemezegységet első készülékként a számítógépre az eredetileg hozzá mellékelt kábelével.

Véletlen számok

A videoprocesszor is képes (csaknem) véletlen számokat generálni. Adva van ugyanis az a lehetőség, hogy annak a rasztersornak a sorszámát lekérdezzük, ahol éppen az elektronsugár tartózkodik. Mivel a kép másodpercenként 60-szor újra képződik, és a képernyőn mintegy 300 rasztersor van, a rekesz tartalma másodpercenként mintegy 15000-szer változik. Ekkor ugyan nem valódi véletlen számok generálódnak, mivel a rekesz értéke mindig 1-gyel növekszik. Ez azonban olyan gyorsan zajlik le, hogy ezt a rekeszt lekérdezve leglábbis BASIC-ban garantáltan kiszámíthatatlan véletlen számok képezhetők. A rekesz lekérdező utasítása:

PRINT PEEK (53266)

KLICK-KLACK

```
AZ 5-RE VEGZODO SORSZAMU SOROKAT NEM KELL BEIRNI!

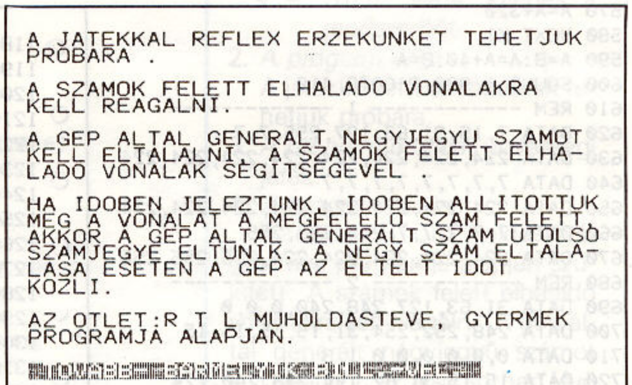
10 REM ****************************
20 REM * KLICK-KLACK              *
30 REM * KESZITETTE:              *
40 REM *       LOOS JOZSEF        *
50 REM * COMMODORE PLUS/4 GEPRE.  *
60 REM ****************************
70 VOL 7
80 PRINT"                "
85 REM [CLR] 5X[LE] 9X[JOBB] [CTR/9] 16X[SPC] [CTR/0]
90 PRINT"   KLICK-KLACK  "
95 REM 9X[JOBB] [CTR/9] 3X[SPC] 2X[SPC] [CTR/0]
100 PRINT"                "
105 REM 9X[JOBB] [CTR/9] 16X[SPC] [CTR/0]
110 PRINT"(SOOLSOFT)":SOUND 2,780,2
115 REM 7X[LE] 12X[JOBB]
120 FOR Q=1 TO 1200:NEXT Q:SOUND 1,890,2
130 PRINT"A JATEKKAL REFLEX ERZEKUNKET TEHETJUK"
135 REM [CLR] 2X[LE] [JOBB]
140 PRINT"PROBARA ."
145 REM [JOBB]
150 PRINT"A SZAMOK FELETT ELHALADO VONALAKRA "
155 REM [LE] [JOBB]
160 PRINT"KELL REAGALNI."
165 REM [JOBB]
170 PRINT"A GEP ALTAL GENERALT NEGYJEGYU SZAMOT"
175 REM [LE] [JOBB]
180 PRINT"KELL ELTALALNI, A SZAMOK FELETT ELHA-"
185 REM [JOBB]
190 PRINT"LADO VONALAK SEGITSEGEVEL ."
195 REM [JOBB]
200 PRINT"HA IDOBEN JELEZTUNK, IDOBEN ALLITOTTUK"
205 REM [LE] [JOBB]
210 PRINT"MEG A VONALAT A MEGFELELO SZAM FELETT,"
215 REM [JOBB]
220 PRINT"AKKOR A GEP ALTAL GENERALT SZAM UTOLSO"
225 REM [JOBB]
230 PRINT"SZAMJEGYE ELTUNIK. A NEGY SZAM ELTALA-"
235 REM [JOBB]
240 PRINT"LASA ESETEN A GEP AZ ELTELT IDOT ."
245 REM [JOBB]
250 PRINT"KOZLI."
255 REM [JOBB]
260 PRINT"AZ OTLET:R T L MUHOLDASTEVE, GYERMEK ":PRINT"PROGRAMJA ALAPJAN."
265 REM [LE] [JOBB] [JOBB]
270 PRINT" TOVABB- BARMELYIK BILL.-VEL. ":SOUND 1,890,2:GETKEY C$:SOUND 2,999,2
275 REM [LE] [JOBB] [CTR/9] [CTR/0]
280 REM --------- JATEKKEP -----------
290 GRAPHIC 1,1
300 CHAR 1,4,16,"1  2  3  4  5  6  7  8  9"
310 FOR I=0 TO 200 STEP 24
320 DRAW 1,36+I,138 TO 36+I,144
330 NEXT I
340 BOX 1,50,20,80,80
350 BOX 1,90,20,120,80:BOX 1,130,20,160,80:BOX 1,170,20,200,80
360 CHAR 1,0,21,"========================================":SOUND 1,890,2
365 REM 40X[=]
370 CHAR 1,26,22,"'M' -ROGZITES"
380 CHAR 1,26,23,"'V' -STOP":SOUND 1,800,2
390 REM ------ VELETELEN ------------
400 V=0:A=9528:B=A
410 V=V+1:IF V=5 THEN GOTO 1240
420 S(V)=INT(RND(1)*9+1)
430 IF S(V)=1 THEN RESTORE 610:GOTO 520
440 IF S(V)=2 THEN RESTORE 680:GOTO 520
```

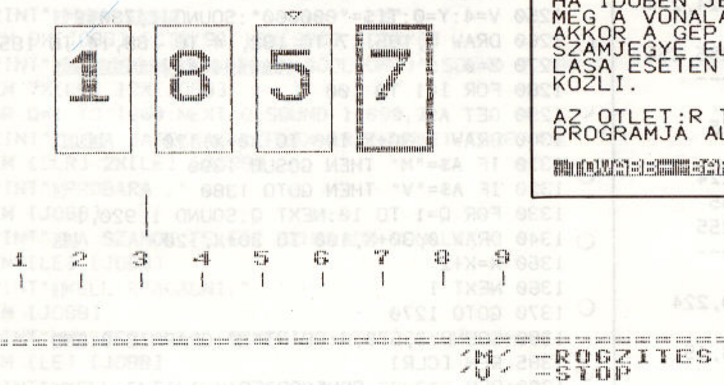
1. *A program kezelése:*
   M – rögzítés, vonal megállítása
   V – STOP, vége, kilépés a rendszerből.
2. *A program célja:*
   A játékkal reflexérzékünket tehetjük próbára.
3. *A program műfaja:* ügyességi játék.
4. *A program leírása:*
   Az ötlet az RTL műholdastévé, gyermek programja alapján született. A számok felett elhaladó vonalakra kell reagálni. A gép által generált négyjegyű számot kell eltalálni, az alsó sorban lévő számok felett elhaladó vonalak segítségével. Ha időben jelzett a játékos, időben állította meg a vonalat a megfelelő szám felett, akkor a gép által generált szám *utolsó* számjegye eltűnik. A négy szám eltalálása esetén a gép az eltelt játékidőt közli a játékossal. Új játék kérésekor, a játék megunásig folytatható.
5. *Programozási feladatok megoldása*
   – a véletlenszám generálása, grafikai megjelenítése
   – játékidő mérése
   – haladó vonal létrehozása, találat érzékelése.

Loós József

```
450 IF S(V)=3 THEN RESTORE 750:GOTO 520
460 IF S(V)=4 THEN RESTORE 820:GOTO 520
470 IF S(V)=5 THEN RESTORE 890:GOTO 520
480 IF S(V)=6 THEN RESTORE 960:GOTO 520
490 IF S(V)=7 THEN RESTORE 1030:GOTO 520
500 IF S(V)=8 THEN RESTORE 1100:GOTO 520
510 IF S(V)=9 THEN RESTORE 1170
520 FOR K1=1 TO 3
530 FOR H=0 TO 15
540 READ L
550 POKE A+H,L
560 NEXT H
570 A=A+320
580 NEXT K1
590 A=B:A=A+40:B=A
600 SOUND 1,980,2:GOTO 410
610 REM ------------ 1 ------------
620 DATA 7,15,31,63,127,255,7,7
630 DATA 224,224,224,224,224,224,224,224
640 DATA 7,7,7,7,7,7,7,7
650 DATA 224,224,224,224,224,224,224,224
660 DATA 7,7,7,7,7,255,255,255
670 DATA 224,224,224,224,224,255,255,255
680 REM ------------ 2 ------------
690 DATA 31,63,127,248,240,0,0,0
700 DATA 248,252,254,31,15,15,15,15
710 DATA 0,0,0,0,0,0,1,3
720 DATA 15,15,31,62,124,248,240,224
730 DATA 7,15,31,62,127,255,255,255
740 DATA 192,128,0,0,255,255,255,255
750 REM ------------ 3 ------------
760 DATA 255,255,255,0,0,0,0,1
770 DATA 255,255,255,30,60,120,240,224
780 DATA 3,7,15,1,0,0,0,0
790 DATA 192,128,224,248,126,31,15,15
800 DATA 0,0,240,240,248,127,63,31
810 DATA 15,15,15,15,31,254,252,248
820 REM ------------ 4 ------------
830 DATA 1,1,3,3,7,7,15,15
840 DATA 248,248,248,248,248,248,120,120
850 DATA 30,30,60,60,120,120,255,255
860 DATA 120,120,120,120,120,120,255,255
870 DATA 255,255,0,0,0,0,0,0
880 DATA 255,255,120,120,120,120,120,120
890 REM ------------ 5 ------------
900 DATA 255,255,255,224,224,224,255,255
910 DATA 255,255,255,0,0,0,248,252
920 DATA 255,0,0,0,0,0,0,0
930 DATA 254,63,31,15,15,15,15,15
940 DATA 0,0,240,240,248,127,63,31
950 DATA 15,15,15,15,31,254,252,248
960 REM ------------ 6 ------------
970 DATA 1,3,7,7,15,15,30
980 DATA 224,192,192,128,128,0,0,0
990 DATA 30,60,63,127,127,248,240,240
1000 DATA 0,0,248,252,254,31,15,15
1010 DATA 240,240,240,240,248,127,63,31
1020 DATA 15,15,15,15,31,254,252,248
1030 REM ------------ 7 ------------
1040 DATA 255,255,255,255,0,0,0,0
1050 DATA 255,255,255,255,30,30,60,60
1060 DATA 0,0,0,0,1,1,3,3
1070 DATA 120,120,240,240,224,224,192,192
1080 DATA 7,7,15,15,30,30,60,60
1090 DATA 128,128,0,0,0,0,0,0
1100 REM ------------ 8 ------------
1110 DATA 31,63,127,248,240,240,240,240
1120 DATA 248,252,254,31,15,15,15,15
1130 DATA 120,63,31,63,120,240,240,240
1140 DATA 30,252,248,252,30,15,15,15
1150 DATA 240,240,240,240,248,127,63,31
1160 DATA 15,15,15,15,31,254,252,248
1170 REM ------------ 9 ------------
1180 DATA 31,63,127,248,240,240,240,240
1190 DATA 248,252,254,31,15,15,15,15
1200 DATA 240,240,248,127,63,31,0,0
1210 DATA 15,15,31,254,254,252,60,120
1220 DATA 0,0,0,1,1,3,3,7
1230 DATA 120,240,240,224,224,192,192,128
1240 REM ------- JATEK -------------
1250 V=4:Y=0:TI$="000000":SOUND 1,780,2
1260 DRAW 1,185,17 TO 182,14 TO 188,14 TO 185,17:SOUND 2,670,1
1270 X=0
1280 FOR I=1 TO 100
1290 GET A$
1300 DRAW 1,30+X,100 TO 30+X,120
1310 IF A$="M" THEN GOSUB 1390
1320 IF A$="V" THEN GOTO 1380
1330 FOR Q=1 TO 10:NEXT Q:SOUND 1,920,1
1340 DRAW 0,30+X,100 TO 30+X,120
1350 X=X+2
1360 NEXT I
1370 GOTO 1270
1380 SOUND 1,670,4:PRINT"♥":GRAPHIC 0:END
1385 REM [CLR]
1390 REM ------ ELLENORZES --------
1400 ON S(V) GOTO 1410,1420,1430,1440,1450,1460,1470,1480,1490
1410 IF X=6 THEN GOTO 1520:ELSE GOTO 1500
1420 IF X=30 THEN GOTO 1520:ELSE GOTO 1500
1430 IF X=54 THEN GOTO 1520:ELSE GOTO 1500
1440 IF X=78 THEN GOTO 1520:ELSE GOTO 1500
1450 IF X=102 THEN GOTO 1520:ELSE GOTO 1500
1460 IF X=126 THEN GOTO 1520:ELSE GOTO 1500
1470 IF X=150 THEN GOTO 1520:ELSE GOTO 1500
1480 IF X=174 THEN GOTO 1520:ELSE GOTO 1500
1490 IF X=198 THEN GOTO 1520
1500 FOR Q=1 TO 700:NEXT Q
1510 RETURN
```

```
1520 BOX 1,172-Y,22,198-Y,78:PAINT 1,171-Y,21
1530 DRAW 0,185-Y,17 TO 182-Y,14 TO 188-Y,14 TO 185-Y,17:Y=Y+40:V=V-1:SOUND 2,450,1
1540 IF V=0 THEN GOTO 1570
1550 DRAW 1,185-Y,17 TO 182-Y,14 TO 188-Y,14 TO 185-Y,17:SOUND 1,900,2
1560 RETURN
1570 REM ------- IDO --------------
1580 RR$=TI$
1590 R1$="IDO":R2$=MID$(RR$,3,2):R3$=MID$(RR$,5,2)
1600 CHAR 1,1,22,R1$:CHAR 1,4,22,"=":CHAR 1,5,22,R2$:CHAR 1,7,22,":"
1610 CHAR 1,8,22,R3$
1620 CHAR 1,1,23,"UJ JATEK (I/N) ?":SOUND 2,780,2
1630 GETKEY J$:SOUND 2,670,3
1640 IF J$="I" THEN GOTO 290
1650 IF J$="N" THEN GOTO 1380
1660 IF J$<>"I" OR J$<>"N" THEN GOTO 1630
```





Tippek—Trükkök
nyomtatókhoz

Hibák az 1571-ben

Ebben a mágneslemezegységben olykor jellegzetes hiba keletkezik: olyan lemezeken, amelyek több mint 50%-ban már megteltek, nehézségek adódnak mind a tárolásnál, mind a betöltésnél. Ennek az oka pontosan abban a kis különbségben keresendő, ami az 1571-et pl. az 1570-től megkülönbözteti. Ez pedig a dupla író/olvasófej. A két fej ugyanis egy kis acélrugóval van egymással összekötve. Ez arra szolgál, hogy a két fej lehetőleg jól érintkezzen a lemez felületével. Ez a kis rugó a gyenge pontja az 1571-nek, könnyen eltörik, és azután a felső fej nem érintkezik többé a lemez felületével. Mivel a felső lemezoldalra csak akkor történik felírás, amikor az alsó megtelt, a félig tele lemez hibajelzést ad. Van segítség ez ellen?

A legjobb a hibát elkerülni. A kis rugó főként akkor van igénybe véve, amikor a lemezegység kapuját lemez behelyezés nélkül lezárják, mert akkor a fej egyoldalú terhelést kap. Emiatt a fej ferdén áll és a rugó megcsavarodik. Ezért a kaput csak akkor zárjuk le, ha a lemezt már betettük (vagy szállításkor a védőkartont behelyeztük).

Centronics-kábel

Egyike a leggyakrabban szükséges hardver eszközöknek. A nyomtatónak a C64 vagy C128 számítógép párhuzamos csatolójával (Centrovics-nak is nevezik) való összekötéséhez szükséges kábel. Ilyen kábelt nagyon egyszerűen lehet készíteni, csupán néhány alkatrész szükséges hozzá:

1 User-Port dugaszoló
2 Centronics dugaszoló
1—2 m 11 eres kábel
Az alábbi érintkezőket kell egymással összekötni:

User-Port	Centrovics
A	16
B	11
C	2
D	3
E	4
F	5
H	6
J	7
K	8
L	9
M	1

C-64 bővítések

Hang digitalizáló – a beszélő C64

A sorozat korábbi számaiban számos C64 bővítést ismertettünk. Ebben a fejezetben a címnek megfelelően beszélni tanítjuk meg a gépet.

Ez a terület egy kicsit misztifikált oldala a számítástechnikának, abban az értelemben, hogy a hiedelem szerint bonyolult és drága áramkörök szükségesek hozzá. A Kedves Olvasó ha végigolvassa ezt a cikket rájön, hogy ez korántsem igaz, mert egyszerű eszközök segítségével is jó eredményt érhetünk el. Természetesen az egyszerű eljárásoknak megvannak a korlátai, ami az alkalmazhatóságnak is határt szab. Ennek ellenére mindig érdemes egy problémát az egyszerű oldaláról megközelíteni, mert a továbblépés lehetősége a kis lépések taktikájának értelmében mindig világosabb, biztonságosabb.

A beszéd, a hang feldolgozásának két változatával foglalkozunk. Az elsőnél a hangforrásból származó jelet (beszéd, zene stb.) digitalizáljuk, eltároljuk, majd visszahallgatjuk. A második esetben valamilyen hangforrás jelét a gép érzékeli és ennek hatására valamilyen feladatot elvégez (pl. zajérzékeléssel működő betörésjelző).

A két változat egymástól teljesen különböző megoldást kíván, hogy mégis egyszerre tárgyaljuk, technológiai oka van. Mindkét áramkör a joystick csatlakozón keresztül kapcsolódik a géphez. A digitalizáló a 2. porton, a hangkapcsoló pedig az 1-es porton keresztül egy-egy 9-pólusú Canon csatlakozó segítségével. Ehhez úgy rögzítjük a panelt, hogy azt élével a csatlakozó két lábsora közé bedugjuk és a fólioldalon hozzáforrasztjuk, így a géphez illesztve a csatlakozó tartja a panelt. Mivel az áramkört a gép védelme érdekében célszerű bedobozolni, doboznak pedig legolcsóbb egy műanyag cartridge dobozt használni, a panel mérete adott. Erre a méretre mindkét áramkör kényelmesen rátervezhető, így a két csatlakozó már stabilan tartja a kártyát. Ezáltal két legyet ütöttünk egy csapásra. Spóroltunk egy dobozt és egy nyákot, valamint stabilan megoldottuk a kártya rögzítését.

A digitalizáló működése

Az áramkör rajza az 1. ábrán látható. A felépítése, működése egyszerű, könnyen érthető. Fontos tulajdonsága, hogy nem igényel külön tápot, azt a joystick portról kapja, így +5 V-ról üzemel. A TTL áramköröknél ez természetes, az LM324 pedig még hajlandó működni ezen a feszültségen.

A bemenet kis- és nagyszintű jeleket fogad. A kisszintű jel az A_1-re kerül, ami amellett, hogy erősít, beállítja az A_2—A_4 komparátorok munkapontját a P_1 segítségével. A komparátorok referencia szintjét R_5—R_8 állítja be. A komparátorok kimenetének szintje a bemenő jel függvényében így alakul:

A_2	A_3	A_4	TTL kimenet	
1	1	1	0	0
1	1	0	0	1
1	0	0	1	0
0	0	0	1	1

Az N_1—N_3 invertálja a komparátorok jelét, N_4—N_7 pedig bináris kóddá alakítja. Innen kerül a jel az R_9, R_{10}-en keresztül a 2. vezérlőportra.

Az áramkör szerelése, élesztése

A panelt ha a *2. ábra* alapján készíti el, ∅1 mm-es fúróval fúrja ki, reszelje méretre, majd erős fénnyel átvilágítva vizsgálja meg, hogy nincs-e rajta zárlat vagy szakadás. Forrassza be az egyetlen átkötő vezetéket, amit az ültetési rajzon folytonos vonal jelöl. Ezután tetszőleges sorrendben forrassza be a többi alkatrészt teljesen a panelra fektetve, hogy a dobozban kényelmesen elférjenek. Utoljára forrassza fel a két Canon csatlakozót. A helyes pozicionálás érdekében ezeket dugja rá a *kikapcsolt (!)* gépre, helyezze a panelt élével a két lábsor közé. Forrasszon be egy-két lábat mindkét csatlakozónál, hogy ezután ne tudjon elmozdulni. Vegye le a gépről az egészet és forrasszon be gondosan minden lábat. Az R_9, R_{10} számára csak egy-egy forrszem van a panelen, a másik lábukat közvetlenül a csatlakozóra kell kötni. A számozás a csatlakozó házon olvasható.

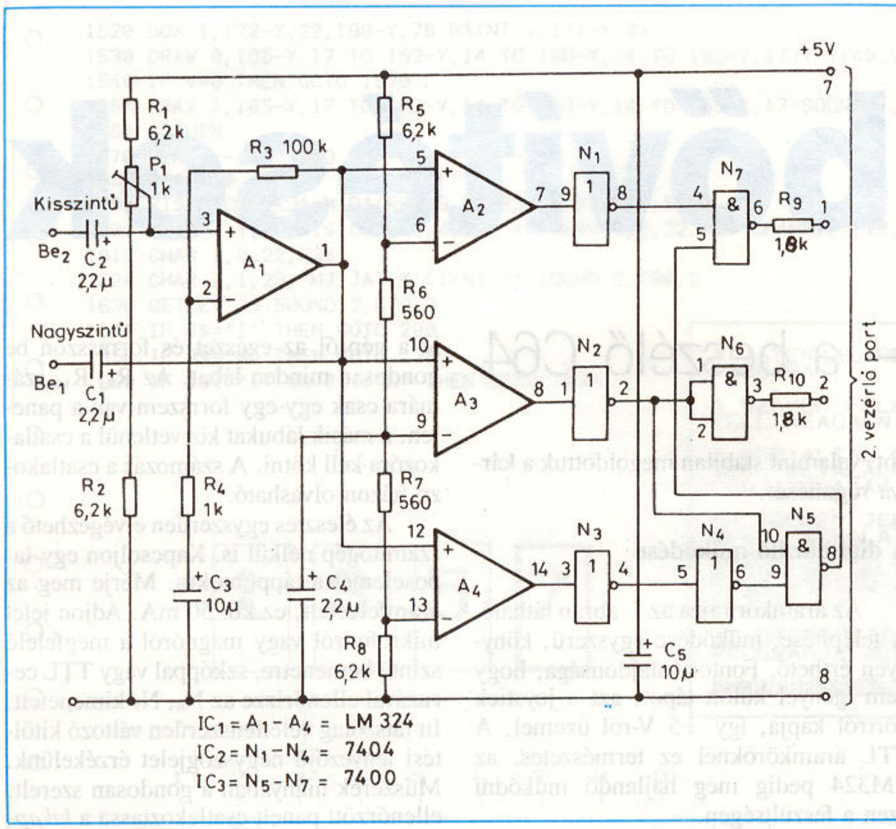
Az élesztés egyszerűen elvégezhető a számítógép nélkül is. Kapcsoljon egy laposelemet a táppontokra. Mérje meg az áramfelvételt, ez kb. 50 mA. Adjon jelet mikrofonról vagy magnóról a megfelelő szintű bemenetre, szkóppal vagy TTL ceruzával ellenőrizze az N_6, N_7 kimeneteit. Itt látszólag véletlenszerűen változó kitöltési tényezőjű négyszögjelet érzékelünk. Műszerek hiányában a gondosan szerelt, ellenőrzött panelt csatlakoztassa a *kikapcsolt (!)* gépre, bekapcsolás után olvassa be a működtető szoftvert lemezről vagy kazettáról. (A szoftvert a szerző által összeállított egységcsomag tartalmazza. Cím a 2. oldalon levő hirdetésben megtalálható.)

A szoftver ismertetőjének elolvasása után készítsen felvételt valamilyen jelforrásból. P_1 segítségével állítsa be a legjobb hangminőséget. A gondosan szerelt áramkörnél is előfordulhat az a hiba, hogy a gépre csatlakoztatva úgy tűnik, megbénítja a billentyűzet működését. Ennek az az oka, hogy alacsony az R_9, R_{10} értéke, és ezek úgy viselkednek a porton, mint a zárlatos joystick. Ilyen esetben meg kell növelni ezek értékét. A rajzon feltüntetett 1,8 kΩ elfogadható értéknek bizonyult.

Beszélő BASIC

A SPEECH (beszélő) BASIC olyan program, amely alkalmassá teszi a C64-et beszéd, zene vagy más analóg jel digitalizálásra.

A program ciklikusan lekérdezi az A/D 2 bitjét és négyes csoportokban bájtokba szervezve eltárolja az értékeket a memóriában. Ott egy monitorszerű programmal változtathatunk rajta, szerkeszthetjük, majd visszajátszhatjuk. Lehetőség van effektek alkalmazására és a lejátszási sebesség szabályozására. A digitalizálás eredményeként kapott adathalmaz lemezre menthető és onnan visszatölthető. A program tág lehetőséget biztosít a kí-



1. ábra

sérletezésre. Ennek megkönnyítése érdekében ismertetem a használható utasításokat, néhány alkalmazási példát, valamint a jellemző adatokat.

A program kb. 4 Kbyte hosszú ($0801=2049-től $17E1 = 6113-ig), mivel használni tudja a ROM-k alatti RAM területeket, így 57,5 Kbyte területen helyezhet el digitalizált adatokat. A mintavételi sebességtől függően 15—45 másodpercnyi anyagot tud letárolni a memóriában. Beszéd esetén, megelégedve a telefon minőséggel, 3 kHz-es felső határfrekvenciát figyelembe véve, 6 kHz-es mintavételi frekvencia adódik. A zenénél 9 kHz-es vágást számolva, a mintavétel 18 kHz lesz.

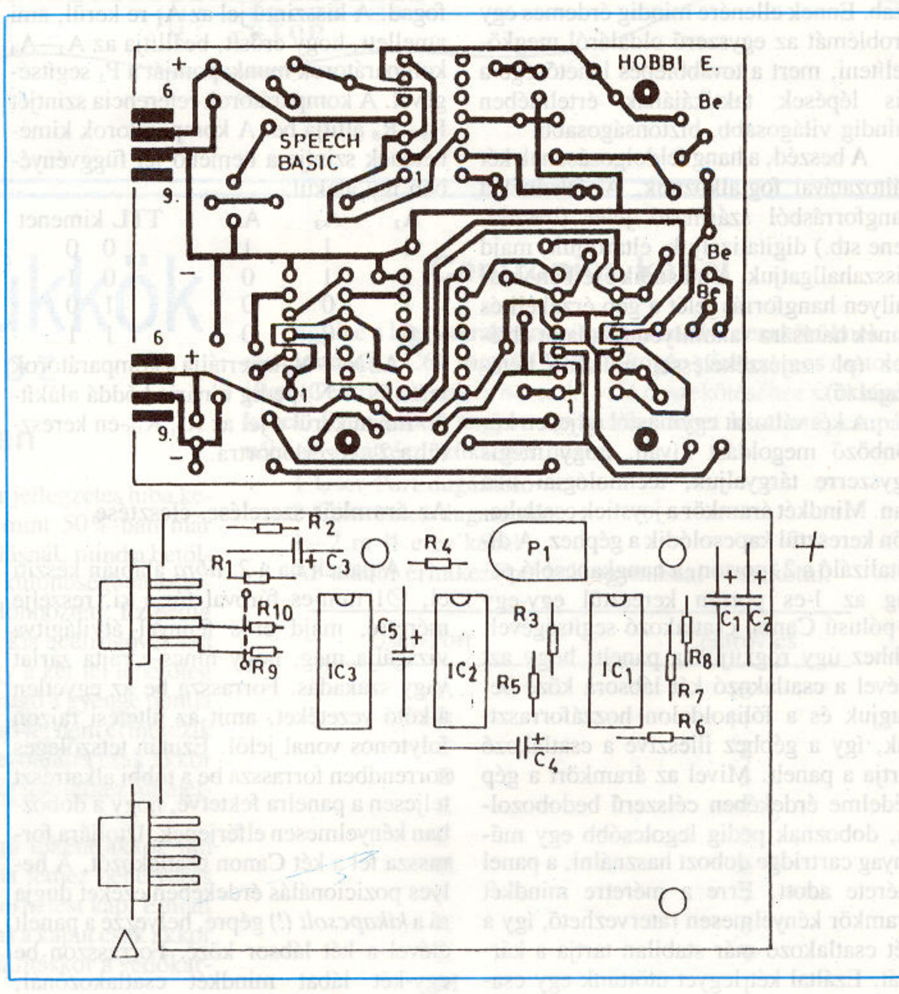
A lejátszó rutin egyszerűen a SID hangerő regiszterébe ($D418) shifteli a 2 bites értékeket. Így alakul ki a hang, amit elég fémessé tesz az, hogy formája négyszögjel. A kísérletező kedvű amatőr esetleg azzal is megpróbálkozhat, hogy saját lejátszó rutint ír, amelyik a jeleket az USER PORT-ra adja ki. Itt ellenálláslétrával történhet a D/A átalakítás, majd a kialakult jelet a SID-be visszavezetve a belső szűrőkkel lehet hűebbé tenni a hangot. Ha már úgyis új lejátszó rutint használunk, érdemes valamilyen tömörítő algoritmuson is elgondolkozni.

Beállítás, behallgatás mód

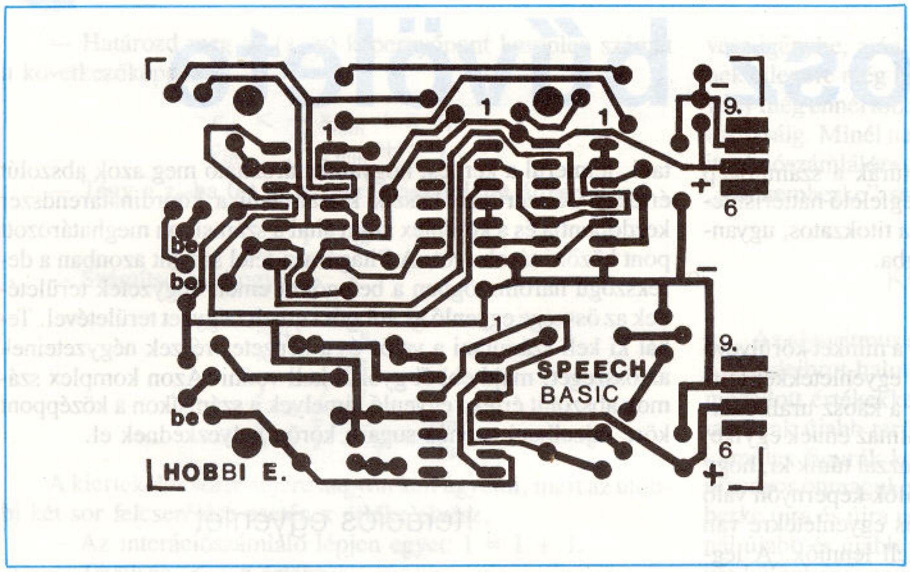
A digitalizált érték közvetlenül a SID-be kerül, nem tárolódik. Így kereshetjük meg magnó használata esetén a kívánt részt, próbálhatjuk ki az áramkört, és beállíthatjuk a trimmer-potenciométert. Működés közben a 2 biten érkező 4 értéknek megfeleltetett színekkel csíkozza a program a képernyőt. Ezek a színek a COL DEF szín 1, 2, 3, 4 utasítással rendelhetők az értékekhez (ajánlott értékek a 0, 11, 12, 15). Ha most a HEAR :SPPEED1 parancsokkal belépünk a behallgatás üzemmódba, „láthatjuk" a hangokat. A trimmerrel úgy kell a komparálási szintet beállítani, hogy a 4 szín egyenletes eloszlású legyen. Kilépés a SPACE gomb megnyomására történik.

Felvétel

Három új utasítást kell megismernünk, hogy felvételt készíthessünk. A HIMEM $ cím a BASIC RAM végét állítja be (alapértéke $A000). Ha maximális felvétel hosszt szeretnénk beállítani, akkor a HIMEM $2000 parancsot adjuk ki. A



2. ábra



3. ábra

memória felosztása a MEM paranccsal kérhető le. A RECORD parancs jelenti a felvételt. Paraméter megadása nélkül a digitalizálás a teljes memórioába történik. A folyamat SPACE megnyomására indul és RESTORE-ral szakítható meg, vagy a memória betelése után automatikusan leáll. Megadható azonban egy terület is a RECORD FROM $ kezdőcím TO $ végcím SPEED X paranccsal, ahol X a mintavételi sebesség (a kezdő és a végcímnek HIMEM-nél nagyobbnak kell lennie).

Lejátszás

A PLAY parancs a tárolt adatok lejátszására szolgál. Szintaxisa megegyezik a RECORD-dal, azaz itt is lehet a lejátszandó területet és a sebességet szabályozni. A RESTORE gomb megnyomása azt a memória címet adja meg, ahol éppen a lejátszás tartott. Ezeket a címeket használhatjuk ún. blokkok kijelölésére a BLOCK X FROM kezdőcím TO végcím megjegyzés felépítésű parancsban, ahol X = 1—31 és a megjegyzés 8 karakter hosszú lehet.

Az utolsó két blokk külöleges célra van lefoglalva. A 30-as funkció billentyűkhöz rendelt parancsokat tartalmazza (lásd később), míg a 31 a blokkhoz rendelt memória területek összekapcsolását mutatja a blokk számokkal. Mielőtt a blokk műveletekre rátérnék még két megjegyzést tennék. Az első, hogy lejátszásnál lehetőség van a digitalizált értékek és a hangerő egymáshoz rendelésére. Erre a VOLDEF hangerő 1, 2, 3, 4 parancs szolgál. A szokásos értékeke 0, 5, 10, 15, azaz egyenletes hozzárendelés. A másik közbevetésem a sebességre vonatkozik. Felvételnél ez 1—40-ig terjedhet, lejátszásnál 0—40-ig és ebben az esetben fordított a hozzárendelés.

Blokkműveletek

A blokktáblát törölhetjük a FOR X = 1 TO 29:BLOCK X " ":NEXT X utasítás sorral. Ha van definiált blokkunk, esetleg több is, a blokktáblát a MAP X—Y utasítással nézhetjük meg, ahol az X-től Y számú blokkig lesz kiírva. A BLOCK utasítás megfelel a MAP 0—31-nek, azaz a teljes táblát listázza. A blokkok minden olyan helyen használhatók, ahol a FROM … TO … utasítást alkalmazzák.

Effektek

Egy blokk ismétlésére használhatjuk a RAP parancsot. Visszhangot az *1. példaprogram* segítségével állíthatunk elő.

File-műveletek

Az adatok lemezre mentése a BSAVE, visszatöltése a BLOAD utasítással történik, a következő formában: BSAVE v. BLOAD „név", egységszám, FROM kezdőcím TO végcím. Természetesen a terület kijelölése itt is történhet blokk hozzárendeléssel.

Egyéb lemezműveletek

A DIR parancs a tartalomjegyzék képernyőn történő megjelenítésére szolgál. DISK parancs string utasítással pedig egy DOS parancs adható ki a floppynak. A file-ok a CLOSE utasításon kívül a RUN STOP és RESTORE billentyűk együttes megnyomásával is lezárhatóak.

Egyéb parancsok

Minden olyan helyen, ahol címet adunk meg, nemcsak decimális szám adható be, hanem a $ jel alkalmazásával hexadecimális és % jel után bináris is. Ha THEN utasítást alkalmazunk, az utána lévő utasítás előtt is kell a kettőspont, mint elválasztójel.

A HELP parancs kiadásával elérhetjük, hogy a program a képernyőre listázza a SPEECH BASIC utasításait. A HELP* ugyanezt teszi az alap BASIC szavakkal.

Az EXEC kulcsszó arra szolgál, hogy mögötte egy stringben több utasítást adhatunk meg rövidítve, összevonva. Ez az adatok miatt leszűkített BASIC területre gondolva előnyös.

A KEY (n string) key def. parancs az n-ik funkcióbillentyűhöz rendeli a stringben megadott parancssort. A KEY parancs a hozzárendeléseket listázza ki.

A PAUSE A utasítás A/60 másodpercig várakozik. A bexadecimális és a decimális szmok átszámítását a HEX, ill. DEZ parancsok segítik.

A képernyő ki/be kapcsolását végzi a SCREEN X utasítás, ahol X=0 a kikapcsolás és X≠0 a bekapcsolás.

A RESET paranccsal alaphelyzetbe hozható a SPEECH BASIC, míg a BASIC utasítás hatására visszatérhetünk az alap BASIC-be.

A MON (block) — parancssor begépelésével egy speciális monitor programba jutunk, amely sgítségével megnézhetjük az adatokat a memóriában és módosíthatjuk is azokat (a módosítást RETURN-nel zárhatjuk le). A blokk helyett a FROM … TO szerkezet is használható.

Példaprogramok:

1. Visszhang:
   PLAY BLOCK X SPEED Y
   FOR I = 5 TO 0 STEP — 5
   VOL DEF 0, 1, 2*I, 3*I
   PLAY BLOCK X SPEED Y
   NEXT I
2. Az EXEC használata:
   PLAY BLOCK 1 SPEED
   PLAY BLOCK 2 PLAY BLOCK 2
   PLAY BLOCK 3
   helyett: EXEC "S1P1P2P3"
3. Terület-kijelölések:
   … FROM $A000
   … TO $D000
   … FROM $A000 TO $D000
   … BLOCK 1

A digitalizálót működés közben megtekintheti, egységcsomag vagy készre szerelt formában megvásárolhatja a szerző üzletében. Cím a 2. oldalon levő hirdetésben megtalálható.

A káosz bűvölete

A komplex számsíkon kialakuló struktúrák a számítógép feltalálása óta szorgos kutatások témáját. Megfelelő háttérismeretekkel a C64-gyel is be lehet hatolni ebbe a titokzatos, ugyanakkor rendkívül bonyolult struktúrájú világba.

*

Azzal a feltételezéssel ellentétben, hogy a minket körülvevő világ jelenségei matematikai képletekkel és egyenletekkel leírhatók, kimutatható, hogy sok dolog mögött a káosz uralkodik. Az „almaemberke", illetve a Mandelbrot-halmaz ennek egyike, és más dinamikus rendszerekhez hasonlóan azzal tűnik ki, hogy rekurzióval leírható és ábrázolható. A fraktálok képernyőn való megjelenítéséhez ezért úgynevezett iterációs egyenletekre van szükségünk, amiknek nem-lineárisaknak kell lenniök. A legegyszerűbb iterációs egyenletet egykor Benoit B. Mandelbrot, az IBM New York-i kutatóközpontjának tudományos munkatársa javasolta. Ez a következő:

$$z_{n+1} = z_n^2 - c$$

Ezek szerint úgy kapjuk meg az iteráció következő tagját, hogy az előző tagot négyzetre emeljük, és ezután kivonunk belőle egy komplex számot. Mivel általában a komplex számok és azok műveletei a számítógépekben nincsenek implementálva, a komplex szám valós (reális) és képzetes (imaginárius) részét külön kell kiszámítani. Mivel nem mindenki ismeri a komplex számokkal való műveletek mikéntjét, a legfontosabb alapismereteket összefoglaljuk.

A komplex számot a legegyszerűbben két szám összegeként állíthatjuk elő a következőképpen:

$$x + i \cdot y$$

Itt az első számot a komplex szám valós részének, a másodikat pedig a képzetes részének, az „i"-t pedig képzetes (imaginárius) egységnek nevezzük. (A komplex algebra szerint $i = \sqrt{(-1)}$. A komplex számokat egyértelműen hozzárendelhetjük a Gauss-féle számsík egy pontjához.

Ha például az 5 + i.4 számot ábrázoljuk a számsíkon, a Descartes-koordinátarendszer kezdőpontjától kindulva az x-tengelyen öt egységet, az y-tengelyen négy egységet lépünk.

Képzetes és valós számok

A komplex számok összeadása és kivonása egyszerű szabályok szerint történik. Külön-külön adjuk össze, illetve vonjuk ki az összeadandók valós és képzetes részét, így:

$$(4 + i.5) + (12 - i.7) = (16 - i.2)$$

A számsíkon ez két vektor összeadásának felel meg. A szorzás azonban már kissé bonyolultabb. Például a

$$(3 + i.6) \cdot (2 - i)$$

számok kiszorzása után eredményül kapjuk:

$$6 - i.3 + i.12 - 6 \cdot i^2.$$

Mivel azonban a definíció szerint $i^2 = -1$, az eredmény összevonása után (12 + i.9). Ezúttal szemmelláthatóan a vektor bizonyos szöggel való elforgatásáról és hosszának megváltozásáról van szó. Akkor, ha komplex számokat össze akarunk hasonlítani, felmerül a kérdés, hogyan határozható meg azok abszolút értéke. Célszerűnek látszik kiszámítani a koordinátarendszer kezdőpontja és a komplex szám által a számsíkon meghatározott pont közötti távolságot. A Pitagórasz-tétel szerint azonban a derékszögű háromszögben a befogókra emelt négyzetek területének az összege egyenlő az átfogóra emelt négyzet területével. Tehát ki kell számítani a valós és a képzetes részek négyzeteinek az összegét, majd ebből gyököt kell vonni. Azon komplex számok abszolút értéke egyenlő, amelyek a számsíkon a középpont köré rajzolható azonos sugarú körön helyezkednek el.

Iterációs egyenlet

Ezzel rendelkezésünkre áll az iterációs egyenletünkhöz szükséges alapvető számítási mód. A

$$c = c_{re} + i \cdot c_{im} \text{ és } z = x + i \cdot y$$

egyenletekkel az elsőnek felírt egyenletünket így írhatjuk:

$$z_{n+1} = x_{n+1} + i \cdot y_{n+1} = x_n^2 - y_n^2 - c_{re} + i \cdot (2 \cdot x_n \cdot y_n - c_{im})$$

Ebben négy egymástól független paraméter van, nevezetesen x, y, c_{re} és c_{im}.

Ezekből kettőt meghatározunk a képernyőn a pont (pixel) helyének a koordinátáival, azaz minden képponthoz egy-egy komplex számot rendelünk hozzá, ami a c_{re} és c_{im} összetevők kezdőértéke. A másik két paramétert esetünkben nullára állítjuk azaz z = 0. Ez fordítva is megy: egy tetszőleges „c" komplex számot az iteráció során állandónak (konstansnak) veszünk, és „z" összetevői lesznek egyenlők a képernyő-koordinátákkal. Ily módon újból komplex alakzat képződik, amit Julia-halmaznak nevezünk.

Ha tehát programot akarunk írni, a képernyőhöz a komplex számsík egy területét kell hozzárendelnünk. Ehhez négy változó szükséges, amiket nevezzünk így: x_{min}, x_{max}, y_{min}, y_{max}. A képernyőnk felbontásán túlmenően szükségünk van még a két képernyőpont (pixel) közötti távolságra x és y irányban. A C64 inicializáló rutinja tehát a következő lesz:

$$x_f = 320;$$
$$y_f = 200,$$

ahol az „f" index a felbontásra utal.

A valós és képzetes tengely terjedjen ki pl. —2-től 2-ig, tehát

$$x_{min} = -2; x_{max} = +2;$$
$$y_{min} = -2; y_{max} = +2;$$

Két szomszédos pont egymástól való távolsága:

$$x_t = (x_{max} - x_{min})/x_f$$
$$y_t = (y_{max} - y_{min})/y_f,$$

ahol a „t" index a távolságra utal.

Ezzel az iterációs egyenlettel keressük most mindazon pontokat, amelyek a Mandelbrot-halmazokon belül esnek. Ezúttal komplex számokról van szó, amelyek tetszőleges számú iteráció után is még végesek maradnak, más esetben a végtelenbe divergálnak. Úgy is mondják, hogy a Mandelbrot-halmazok „két attraktorból" állnak. Ezért minden képpont (a C64 képernyőjén 64000 van!) végig kell fusson a következő algoritmus:

— Határozd meg az (x, y) képernyőpont komplex számát a következőképpen:

$$c_{re} < -x_{min} + x \cdot x_t;$$
$$c_{im} < -y_{min} + y \cdot y_t.$$

— Tégy a z_0-ba 0-t és az iterációszámlálót állítsd 0-ra:

$$I = 0.$$

— Számítsd ki, hogy

$$z_{n+1} \leftarrow z_n - c$$
$$x^2 = x \cdot x;$$
$$y^2 = y \cdot y;$$
$$y = 2 \cdot x \cdot y - c_{cim};$$
$$x = x^2 - y^2 - c_n.$$

A kiértékelés sorrendjére nagyon kell ügyelni, mert az utóbbi két sor felcserélése esetén x értéke elvész.

— Az interációszámláló lépjen egyet: $I = I + 1$.

— Távolság $< -z$ értéke.

— Távolság $= x^2 + y^2$, amíg a távolság <100-nál és az interációszámláló <150-nél.

— Világítsd ki az (x, y) képpontot a (számláló)-nak megfelelő színnel.

Itt most abból a tulajdonságból húzunk hasznot, hogy a 100-nál nagyobb komplex számok a végtelenbe divergálnak. Az algoritmus befejezéséhez egy fix pont esetében szükségünk van a kiugrási feltételre, amit az interációszámlálóval valósítunk meg. A procedúra elhagyása után minden képponthoz olyan színt rendelünk hozzá, ami a számláló értékének felel meg. Mivel többnyire nem áll rendelkezésre elegendő szín, beosztjuk a színeket periodikusan egy mezőbe. Például három szín esetén a választék legyen a következő:

1. szín = piros;
2. szín = sárga;
3. szín = kék;
4. szín = piros;
5. szín = sárga... és így tovább.

Véges attraktor esetén a képernyőpont feketére színeződik.

Mivel az ilyen képek előállítása nagyon sok számítási időt vesz igénybe, ajánlatos a programot gépi kódba realizálni. Ennek ellenére még így is sok türelemre lesz szükség, mert 24 óra vagy még ennél több is eltelik a meghatározott elképzelés megvalósításáig. Minél nagyobbra választjuk a távolság-változóra és az iterációszámlálóra a kilépési feltételt, annál pontosabb lesz az „almaemberke” struktúrája.

Komplex struktúrák

Az algoritmus segítségével abba a helyzetbe kerülünk, hogy a Mandelbrot-halmazokat kutassuk. Hozzuk létre például a fent megadott értékekkel az első képet, tároljuk lemezre, majd válasszunk újabb területet. Különösen a halmazok határán olyan komplex figurák keletkeznek, amelyek az eredetiekhez képest bizonyos önmagukhoz való hasonlóságot mutatnak. Az almaemberke újra és újra nagyításokon esik át, de pontosabb vizsgálatnál újabb és újabb részletek derülnek ki. Ennek következtében két különböző, egyébként hasonló objektum sohasem lesz azonos. Egy további meglepő tulajdonság azt mutatja, hogy az almaemberke összefüggő, így tehát a Mandelbrot-halmazon belül nincsenek olyan pontok, amelyek a mi iterációs eljárásunkkal a végtelenbe igyekeznének.

A legjobb, ha begépeljük a BASIC-listát, hogy vele a fraktál alakzatok kezelését illetően az első tapasztalatokat megszerezzük.

Röviden és velősen

A Mandelbrot-halmazok kiszámításához matematikai előismeretek egyáltalán nem hátrányosak, sőt kifejezetten előnyösek. A képernyő a komplex számsíknak csupán egy darabkáját ábrázolja tulajdonképpen. Minden egyes képpont, akár ki van gyújtva, akár nincs, egy-egy komplex számot képvisel. Annak valós része az x-tengelyen, képzetes része az y-tengelyen van. A C64 felbontóképessége közismerten 320×200 képpont HiRes módban, míg csupán 160×200 pont Multicolor-HiRes módban. Összesen tehát 64000, illetve 32000 címezhető képpont van a képernyőn. Mindegyik képponthoz a rekurziós képlettel kons-

```
1 Poke53281,0:Print"■■ fraktal hegyek ■":Print"■demo":9=53272
2 Poke53280,0:inPut"max magas(32-80)";h:ifh<32orh>80thenPrint"■";goto2
3 fori=1to4:Print"■magas"i":";-h;" - "h;:inPuth(i):h(i)=h(i)+128:nexti:h=h+128
4 inPut"faktor(0-1):";k:for i=16500 to 20724:Pokei,255:next:Poke16500,h(1):eP=0
5 Poke 16564,h(2):Poke20660,h(3):Poke20724,h(4):Print"szamolok■"
6 fori=6to1 steP-1:s=2↑i:fory=0to63stePs:forx=0to63stePs:ad=16500+x+y*65+s*33
7 j=s*33:z1=Peek(ad-j):z2=Peek(ad+j):j=j-s:z3=Peek(ad-j):z4=Peek(ad+j)
8 z=int((z1+z2+z3+z4)/4+rnd(1)*2*s+1-s)+1:ad(5)=ad-s*32.5:ad(6)=ad+s*32.5
9 Print"■";int(eP)"■ ber.":ifabs(z-128)<(h-128)thenz=z-s/2*sgn(z-128)
10 z(5)=int((z1+z+z3)/3+k):z(6)=int((z4+z+z3)/3+k):z(7)=int((z3+z+z2)/3+k)
11 z(8)=int((z(4)+z+z1)/3+k):eP=eP+.07326:Pokead,z:ad(7)=ad+s/2:ad(8)=ad-s/2
12 forq=5to8:c(q)=Peek(ad(q)):ifc(q)<>255andc(q)<>0thenz(q)=(c(q)+z(q))/2
13 ifabs(z(q)-128)>(h-128)thenz(q)=z(q)-s/2*sgn(z(q)-128)
14 Pokead(q),int(z(q)):next q,x,y,i:Poke53272,29:Poke53265,59:Poke53270,248
15 fori=0to999:Poke1024+i,213:Poke55296+i,102:next:fori=8192to16191:Pokei,0
16 next:fory=0to64:forx=0to64:ad=16500+y*65+x:s=Peek(ad):ifs<128thens=128
17 s=s-128:u=2*(x+y+13):ts=s-5*sgn(s):fort=tstos:v=y+90-t:b=int(u/8)*8:bt=6+b-u
18 a2=8192+(vand248)*39+b+v:d=2:w=t:P=Peek(a2)and(255-3*2↑bt):ifs=0thend=3:w=0
19 d=d-int((wand3)/3):Pokea2,Pord*2↑bt:nextt,x,y:Poke198,0:wait198,1:Poke9,21
20 Poke53265,27:Print"■■P043,0:P044,32:P045,0:P046,64■";:Poke631,13:Poke198,1

ready.
```

tansként komplex számot rendelünk hozzá. Az iterációs lépések bizonyos száma után a program felismeri, hogy a következő tagok a végtelenbe igyekeznek, és akkor a pontot az adott koordinátához hozzárendeli. Ha azonban a következő tagok még határozatlanok, egyszerűen feltesszük, hogy a pont a Mandelbrot-halmazokhoz tartozik.

További információk

Végül az érdeklődő olvasónak azokból az irodalmi hivatkozásokból adunk válogatást, amelyek a tárgyalt témával bővebben foglalkoznak. Ezekben a fraktális képek létrehozásához szükséges különböző (pl. a Newton-, a fügefa- és a többi) iterációs eljárások mellett tárgyalnak más ábrázolási módokat)gömbprojeció, háromdimenziós tájképek és így tovább) és azok programozását is.

IRODALOM

1. The Beauthy of Fractals (A fraktálisok szépsége)
PEITGEN, H.–O., RICHTER, P.; Springer Verlag.
2. Die unendliche Reise (A végtelen utazás)
PEITGEN, H.–O., RICHTER, P.; Geo Nr. 7, 1985 július
3. The Science of Fractal Images (A fraktál képek tudománya)
PEITGEN, H.–O. SAUPE, D.; Springer Verlag.
4. Dynamische Systeme und Fraktale. Computergrafische Experimente mit Pascal (Dinamikus rendszerek és fraktálisok. Kísérletek Pascalban)
BECKER, K. H., DÖRFLER, M.; Vieweg Verlag
5. Die Entdeckung des Chaos. Eine Reise furch die Chaos-Theorie (A káosz felfedezése. Utazás a káoszelméletben)
BRIGGS, J., PEAT, F. D.; Carl Hanser Verlag
6. Spektrum der Wissenschaft (A tudomány spektruma)
DEWDNEY, A. K.; Computer Kurzweil
7. Der Flohmarkt der unbegrenzten Möglichkeiten (A korlátlan lehetőségek bolhapiaca)
PÖPPE, Ch., Spektrum der Wissenschaft, 1991. 7–10. sz.

Resetálló basic

Néhány gépi kódú utasítás segítségével resetálóvá lehet tenni a C128-as BASIC programokat. Adjuk be ehhez a Protect 128 programot. Annak indítása után a BASIC programot még leállítani sem lehet resettel. A fifikás számítógépesek persze arra a gondolatra is jöhetnek, hogy a C128-ason a 64-es módban váltsanak ki resetet. Ebből az üzemmódból azután a Save Bas programmal lehet a programot kimenteni. Ha egy ilyen módon kimentett programot a 128-as módban behívunk a DLOAD „név" utasítással, az editálás és a megtekintés problémák nélkül megtörténhet. Éppen ezért ajánlatos 64-es módot blokkolni. Ehhez a Block 64 bevitele szükséges. Ennek használatakor ez az üzemmód nem lesz elérhető. A BASIC programunkat ezekkel a módszerekkel szinte teljesen levédtük.

```
PROTECT-128

10 a=dec("1300"):read b$
20 b=dec(b$)
30 poke a,b:a=a+1:read b$:if b$<>"*" then 20
40 data a9,00,8d,00,ff,20,84,ff
50 data 20,7a,41,20,12,41,a9,c1
60 data 8d,04,0a,58,4c,a6,5a,*
70 bank1:poke dec("fff8"),0:poke dec("fff9"),19

ready.

Save bas  C-128

1 sys 57812 "basic",8
2 poke 174,0:poke 175,160
3 poke 193,1:poke 194,28
4 sys 62597

ready.

Block 64  C-128

10 a=dec("8000"):read b$
20 b=dec(b$)
30 poke a,b:a=a+1:read b$:if b$<>"*" then 20
40 data 09,80,09,80,c3,c2,cd,38
50 data 30,02,*

ready.
```

A VDC RAM egyszerű írása

Eleddig a BASIC programnyelv lassú volt a VDC RAM fölülírásához vagy kiolvasásához. Egy egyszerű trükkel azonban ennek ellenére lehetséges a VDC RAM-ra hivatkozni a BASIC 7.0-ból. Ehhez fölhasználjuk a C128-as néhány operációs rendszer rutinját. A VDC-Write programmal könnyen felülírhatunk egy tetszőleges helyet, a VDC-Read segítségével pedig onnan olvasni is tudunk. Ezeknél a programoknál a 10. sorban az INPUT pranccsal kérdezzük le a kívánt címet. A VDC-Write esetében még a beírandó értékre is szükség van. A kiolvasásnál az adott értéket kijelezzük.

```
VDC-write  C-128

10 bank15:input "cim,ertek";nr,sz
20 sysdec("cdcc"),nr/256,18
30 sysdec("cdcc"),nr and 255,19
40 sysdec("cdca"),sz

ready.

VDC-read  C-128

10 bank15:input "cim";nr
20 sysdec("cdcc"),nr/256,18
30 sysdec("cdcc"),nr and 255,19
40 sysdec("cdd8"):rreg sz:print sz

ready.
```

Még többet ésszel!

XIV. rész

Foglalkozzunk tehát először a magnóval, vagy ahogy még hívják, a Datasette-tel! Ha a spéci csatlakozót nem nézzük, akkor ez egy majdnem közönséges magnó, amelynek a motorját a számítógép tudja elindítani és megállítani. A magnószalagra a felvevőfej úgy rögzíti az adatokat (persze most leegyszerűsítem a dolgot), hogy minden 1-es bit helyére egy magas, minden 0-s helyére egy alacsony hangú jelet ír. A felvett anyag közönséges magnón is visszahallgatható. Mivel a jelek nagyon sűrűn sorakoznak a szalagon, ezért nem egy fütyörészést, hanem egy borzasztó, recsegő zajt hallunk.

Magnót háttértárnak ma már csak azok használnak, akiknek nincs pénzük egy lemezegységre, amely összehasonlíthatatlanul jobb eszköz. A magnó legfontosabb hátrányai a következők:

1. A magnószalag elég sérülékeny. Ha valahol megnyúlik vagy meggyűrődik, ott a felvett jel eltorzul. Ha ez a torzulás meghalad bizonyos mértéket, akkor az adatok beolvasása hibaüzenettel félbeszakad.

2. Lassú. Ez azért van, mert a jelek a lemezhez viszonyítva nagyobb távolságra vannak egymástól, megnövelve a jelek méretét. Így csökkenthető a szalag mechanikus sérüléseinek káros hatása. Persze használhatunk turbóprogramot (ld. később), de ez csak programfile-okra működik, és növeli az olvasási hiba kockázatát.

3. A szalagra csak soros típusú állományok írhatók. Minden állományt minden megnyitáskor a legelején kell elkezdenünk olvasni. Minden byte csak úgy érhető el, hogy az összes megelőző byte-ot is kiolvassuk. Javítani csak úgy lehet, ha beolvassuk és módosítva visszaírjuk a teljes állományt.

4. A file-ok helyét a kezelőnek kell megkeresnie és beállítania. Ez egyrészt kényelmetlen, hiszen ha egy állományt valamiért újra el kell olvasni, akkor a kazettát vissza kell rá állítani. Másrészt nem biztonságos, mert nem nagyon lehet ellenőrizni, hogy jó helyre van-e a kazetta állítva, és igen könnyű így véletlenül felülírni egy másik állományt.

Szalagon kétfajta állomány rögzíthető: program és adatállomány. Mindkettő soros típusú, de eltérő célra használjuk őket. Programfile a **SAVE [fn[,dn[,sa]]]** paranccsal (esetleg utasítással) hozható létre. Az ,,fn'' a filenév rövidítése, és egy karakteres kifejezés lehet, akár 0 byte hosszú is. Az eszközszámot (dn) elhagyva a gép az 1-es egységet, azaz a magnót tekinti célnak. A másodlagos cím (sa) 0 és 255 között bármi lehet, de csak három érték van dokumentálva. A ,,0'' és ,,1'' értékek mentéskor a tapasztalatom szerint egyenértékűek, és közönséges mentést jelentenek. Az ,,sa'' helyére ,,2''-t írva a gép a mentés befejezése után egy ,,Szalag Vége'' jelet (angol rövidítéssel EOT) is a szalagra ír. Ez egy kb. 15 másodperc hosszú különleges fejléc-blokk, amely elolvasásakor a gép a ,,Device not present'' hibaüzenettel áll le. Ilyenre azért lehet szükség, hogy ha a szalagon a betöltéskor megadott nevű programot a gép valamiért nem találja meg, akkor ne olvassa végig hiába az egész szalagot. Természetesen ha a kazettát az EOT jelzés utánra állítjuk, akkor onnan zavartalanul olvashatunk a kazetta végéig vagy a következő EOT jelig.

Nemcsak BASIC program menthető ki. A gépben a BASIC által használt tár különböző szakaszainak címét megtalálhatjuk néhány byte-ban. A TXTTAB nevű bytepár ⟨$2B/2C⟩ mutat a BASIC program elejére, alsó/felső byte sorrendben, a változók területének kezdőcíme a VARTAB nevű mutatóban ⟨$2D/2E⟩ található, értéke rendesen nagyobb az előzőnél. A SAVE utasítás mindig e két mutató közötti területet menti ki, a VARTAB által jelzett byte-ot már nem beleértve. Tehát a mutatók átállításával kimenthetünk gépi kódú programot, karakterkészletet, grafikus hátteret, bármit. A kimentett szakasz kezdőcímét a gép szintén a szalagra írja.

Program betöltésére a **LOAD [fn[,dn[sa]]]** parancs szolgál. A másodlagos címnek itt más jelentése van. Ha értéke ,,0'' (vagy nem írtuk ki), akkor a programot a TXTTAB mutató pillanatnyi értékétől kezdve tölti a tárba, majd megkísérli a BASIC programsorok jellegzetes, egy későbbi részben bemutatandó láncolását helyreállítani. Ennek előnye, hogy egy BASIC program nemcsak azon a társzakaszon futtatható, ahol megíráskor elhelyezkedett. A mutatók ügyes állítgatásával egyszerre akár több BASIC program is a tárban tartható. Az újraláncolás hátránya, hogy ha nem BASIC programot töltünk be így, akkor azon a gép káros változtatásokat végez. Ilyen programfile-ok betöltésekor az ,,sa''-nak nullától különböző értéket kell adni, ettől a gép a programot változtatás nélkül tölti arra a címre, ahonnan ki lett mentve, függetlenül a TXTTAB értékétől.

Szakállas téma, de nem biztos, hogy mindenki által ismert a LOAD működése utasításként, tehát program-módban. A legfontosabb, hogy a betöltés után elölről indul a TXTTAB-nál kezdődő BASIC program, de a korábban felvett változók értékei nem törlődnek. Ha a betöltött program BASIC nyelvű, akkor az indul el, ha valami más helyre való kiegészítő file (és ,,sa'' > 0 paraméterrel töltjük be), akkor a régi indul újra. Ez utóbbi esetben végtelen ciklus lenne a mutatvány eredménye, ha nem gondoskodnánk arról, hogy a töltés befejeztével a LOAD után folytatódjon a futás. Ehhez kihasználható, hogy a változók nem törlődnek. Mutatok három egyszerű példát, amelyekben nem foglalkoztam azzal, hogy mi a teendő, ha a betölteni kívánt file-t a gép nem találja, vagy olvasásakor hibát tapasztal. Az ilyen finomságokat nem biztos, hogy magnóval rendesen meg tudnám oldani, mert nagyon kevés róla a tapasztalatom, másrészt most nem is lenne jó ilyenbe bonyolódni.

Az első példa akkor használható, ha a programot azzal akarom kezdeni, hogy betöltök egy kiegészítő file-t, és aztán a főprogram fut. A hibátlan működés feltétele, hogy a kulcsváltozó értéke az első indításkor ,,0'' legyen, ami RUN-nal való indítással egyszerűen teljesül. A sort esetleg megelőző utasítások a töltés után megismétlődnek. Ha ezek valamelyike törli a Z változót, akkor a program végtelen ciklusba esik.

A 2. példa azt mutatja, hogyan lehet egy nem a program elején levő LOAD-ot beilleszteni. Itt a 0. sor a program első sora kell legyen (a REM-eket nem számítva), és a LOAD nem lehet szubrutinban vagy FOR-ciklusban.

A 3. példa az 1. változata, de több betöltendő file-lal. A módszerek persze kombinálhatók, fejleszthetők, mindig a pillanatnyi feladathoz igazodva.

Ha valaki több BASIC programból álló csomag rátöltéses (overlay) szervezésén töri a fejét, annak elmondom, hogy BASIC társzakaszokat jelző mutatók nem igazodnak az újonnan betöltött programhoz. Ha az új program hosszabb a réginél, akkor rálóg a változók egy részére is, amelyeket ezután kiolvasni már nem érdemes, átírni pedig nem szabad. Ha az új program a rövidebb, a fennmaradó terület csak a változók áthelyezésével szabadítható

fel. Kezdőknek azt javaslom, hogy az ilyen programok ne használják egymás változó-értékeit.

Korábban említettem az ún. turbókat. Ezek a rendszert bővítő programok egy csoportja. A turbó célja, hogy a szalagra a szokásosnál többször sűrűbben írja fel a kimentett program bitjeit, ezáltal töredékére csökkentve a kimentéshez, de főleg a betöltéshez szükséges időt. Mentéskor a turbót be kell tölteni, elindítani, ezáltal az beépül az operációs rendszerbe. Minden ezután kiadott SAVE parancs, vagy esetleg egy megkülönböztető jel hatására a mentés turbó sűrűséggel történik. Az ilyen formában tárolt programok egy közönséges LOAD utasítással nem tölthetők be, mert a gép nem tudja azonosítani a kapott „hadaró" jeleket, csak egy turbó segítségével. Üzletekben kapható programkazettákon szokásos eljárás, hogy a programcsomag file-jainak egyike egy turbó-betöltő, amelyet az indítóprogram automatikusan behív. Szerencse, hogy a sokféle turbó általában el tudja olvasni a többiek munkáját, ezért nem kell egy ismeretlen módon felvett játékprogram betöltéséhez egy hatalmas turbó-gyűjteményt végigpróbálgatni.

Ha egy programot kimentünk, nem lehetünk biztosak abban, hogy a rögzítés biztonságosan sikerült. Ha megpróbáljuk visszatölteni a programot, és az valahol hibás, akkor elvesztettük a hibátlan eredetit is. Ilyenkor kell használni a **VERIFY [fn[,dn[,sa]]]** parancsot. Ez végigolvassa a tárolt file-t, és összehasonlítja a tárban levővel. Az „sa" paraméter jelentése megegyezik a LOAD-nál leírtakkal. Ha a gép a két program között eltérést talál, a „Verify error" hibaüzenetet kapjuk, de sajnos sehogy nem állapítható meg az eltérés helye.

Eddig csak a programfile-okról beszéltem, pedig a magnó adtok tárolására is alkalmas. Mivel ilyen célra a magnót ritkán szokás használni, és a módszer lényegében egyezik a SEQ típusú lemezes állományok kezelésével, ezért ezzel most nem foglalkozom. Csupán megemlítem a magnó néhány sajátosságát.

```
(1.)
1 IF Z=0 THEN Z=1: LOAD "FILE",1,1

(2.)
0 IF Z THEN 19
...
18 Z=1: LOAD "FILE",1,1
19 ...

(3.)
1 Z=Z+1: ON Z GOTO 2,3,4
2 LOAD "FILE1",1,1
3 LOAD "FILE2",1,1
4 ...
```

Az OPEN-ben az „sa" paraméternek három dokumentált jelentése van. „0": megnyitás olvasásra. „1"; megnyitás írásra. „2": megnyitás írásra, lezárás után EOT jelzés felírása. Az OPEN utasítás „parm" tagja (ld. előző rész) egy karakteres kifejezés, amely a file nevét tartalmazza, hossza 0—16 byte, értéke bármi lehet. Globális filenevek, amelyek a lemezen megszokottak, itt nem használhatók.

Az állomány írásra való megnyitásakor először a nevet, egyéb jellemzőket tartalmazó fejléc kerül a szalagra, aztán sorban az adatok. A gép a magnóra nem közvetlenül ír, hanem az adatokat összegyűjti az ún. szalagpufferban, amelynek címe C64-en $033C. Ha ez a puffer megtelik, akkor azt a blokkot egyben a szalagra menti. Hasonlóan történik az olvasás is. Ennek következtében a magnó használatakor a szalagpuffert más célra (pl. gépi kódú programok elhelyezése) nem helyes használni. Írásra nyitott állomány lezárásakor a szalagra kerül egy „Állomány Vége" (EOF) jelzés, és ha kell egy EOT jelzés is.

Most már végre komoly dolgok jöhetnek, a következő néhány rész a lemezegységről fog szólni.

Hódi Gyula

Ötmezős kono

A meghirdetett pályázatra készítettem ezt a programot. Ez a játÉk a mellékelt lemezen van, ÖTMEZŐS KONO néven. Ketten játszhatják; célja, hogy a bábukat az ellenfél bábuinak helyére vigyük. De bővebben a programban olvashatunk erről. Betöltése: LOAD"*",8-cal és RUN-nal indítható.

A program elején szubrutinok találhatók (a bábuk helyeinek képernyő-memória címei, a tábla rajza, a kiírások). Az 500-as sortól kezdődik a főprogram.

A változók nevei és jelentésük:

JN$ játékos neve
J következő játékos
M1$ annak a bábunak a betűjele, mellyel lépni akarunk
L u.a., csak a száma, ami kiírásokhoz is kell
M2$ ahová lépni akarunk (betű)
K u.a. (száma)
BSZ bábu színe
LSZ lépések száma
BA ahol állhatnak bábuk (képernyő-memóriacím)
F ha áll ott bábu, akkor értéke egyenlő a játükos számával; ha nem áll bábu, értéke 0.

A 200 sortól kezdődő alprogram vizsgálja, hogy a beírt lépés megvalósítható-e. Ha nem, hibaüzenetet ad; ha igen, a bábut Áthelyezi (az F értéke megfelelően módosul).

Remélem megnyeri tetszésüket és díjat nyerek.

Tisztelettel:

Mucsi László

Játék-pályázat

```
10 REM *** OTMEZOS KONO ***
15 REM *** MUCSI LASZLO C-64 ***
17 DIMBA(25),F(25):GOTO500
20 REM --- JATEKSZABALY
25 POKE53281,2:POKE53280,2
30 PRINT"▒▒▒";TAB(5)"OTMEZOS KONO JATEKSZABALYAI:"
35 PRINT"▒▒▒ KET JATEKOS JATSZHATJA. AZ EGYIK"
36 PRINT"▒JATEKOSNAK 7 FEKETE, A MASIKNAK 7 FEHER"
37 PRINT"▒KORONGJA VAN. MINDIG AZ A JATEKOS TESZI"
38 PRINT"▒MEG AZ ELSO LEPEST, AKI FEKETE KORONGGAL"
39 PRINT"JATSZIK. EGY LEPESBEN MINDKET JATEKOS"
40 PRINT"▒CSAK EGY KORONGGAL LEPHET. FELVALTVA"
41 PRINT"▒LEPNEK, ES ELORE, HATRA VAGY ATLOS"
42 PRINT"▒IRANYBAN HALADHATNAK A MEZOKON."
43 PRINT"▒▒▒▒";TAB(16)"<▒SPACE▒>"
44 GETV$:IFV$<>" "THEN44
45 PRINT"▒▒";TAB(5)"OTMEZOS KONO JATEKSZABALYAI:"
46 PRINT"▒▒ A JATEK CELJA A KORONGOKKAL ELJUTNI AZ"
47 PRINT"▒ATELLENES OLDALRA, AZ ELLENFEL KORONGJA-"
48 PRINT"INAK HELYERE. AKI EZT ELSOKENT ERI EL A"
49 PRINT"▒JATEKOSOK KOZUL, AZ A NYERTES."
50 PRINT"▒▒LEPES: ELOSZOR ANNAK A MEZONEK A BETUJE-"
51 PRINT"LET USD BE, AMELYIKEN AZ A BABU ALL,"
52 PRINT"▒AMELYIKKEL LEPNI AKARSZ, UTANA AZET,"
53 PRINT"▒AHOVA AKARSZ."
57 PRINT"▒▒▒    <▒SPACE▒>"
58 GETV$:IFV$<>" "THEN58
59 RETURN
60 REM --- JATEKOSOK SZAMA
70 JSZ=2:PRINT"▒"
72 FORV=1TOJSZ:PRINTV". JATEKOS NEVET KEREM!"::PRINT"(MAX. 10 KARAKTER)"
74 INPUTJN$(V):IFLEN(JN$(V))>10THENPRINT"TUL HOSSZU!":GOTO74
76 NEXTV
78 IFJSZ=1THENJN$(2)="C-64"
80 PRINT"▒▒▒"JN$(1)" FEKETE VAGY FEHER"
82 PRINT"AKARSZ LENNI? (FEKETE/FEHER)"
84 INPUT"▒";V$:IFV$="FEKETE"THENBSZ(1)=1:BSZ(2)=2:RETURN
86 IFV$="FEHER"THENBSZ(1)=2:BSZ(2)=1:RETURN
88 PRINT"▒▒▒":GOTO84
100 REM --- TABLA FELRAJZOLASA
102 POKE53280,.:POKE53281,.:PRINT"▒▒"
104 PRINT"┌───┬───┬───┬───┐":PRINT"|   |   |   |   |"
106 PRINT"|   |   |   |   |":PRINT"├───●───●───●───┤"
108 PRINT"|   |   |   |   |":PRINT"|   |   |   |   |"
110 PRINT"●───●───●───●───●":PRINT"|   |   |   |   |"
112 PRINT"|   |   |   |   |":PRINT"├───●───●───●───┤"
114 PRINT"|   |   |   |   |":PRINT"|   |   |   |   |"
116 PRINT"└───┴───┴───┴───┘":RETURN
120 BA(1)=1064:BA(2)=1069:BA(3)=1074:BA(4)=1079:BA(5)=1084:BA(6)=1184
```

```
122 BA(7)=1189:BA(8)=1194:BA(9)=1199:BA(10)=1204:BA(11)=1304:BA(12)=1309
124 BA(13)=1314:BA(14)=1319:BA(15)=1324:BA(16)=1424:BA(17)=1429:BA(18)=1434
126 BA(19)=1439:BA(20)=1444:BA(21)=1544:BA(22)=1549:BA(23)=1554:BA(24)=1559
128 BA(25)=1564:RETURN
130 FORV=1TO6:F(V)=BSZ(1):NEXT:F(10)=BSZ(1):F(16)=BSZ(2):FORV=20TO25:F(V)=BSZ(2)
131 NEXT
132 FORV=1TO6:POKEBA(V),215:POKE54272+BA(V),BSZ(1)-1:NEXT:
134 POKEBA(10),215:POKE54272+BA(10),BSZ(1)-1:POKEBA(16),215:POKE54272+BA(16),BSZ(2)-1
135 FORV=20TO25:POKEBA(V),215:POKE54272+BA(V),BSZ(2)-1:NEXT:RETURN
140 REM --- KIIRAS
142 PRINT"▒"TAB(25)"▒1. JATEKOS:"SPC(29)"-----------"
144 PRINTTAB(25)"▒"JN$(1)"  ";:POKE646,BSZ(1)-1:PRINT"▒O▒"
146 PRINTTAB(25)"▒LEPESEK SZAMA:"SPC(72);LSZ(1)
148 PRINTTAB(24)"▒-----------------"
149 RETURN
150 PRINT"▒"
151 PRINTTAB(25)"▒2. JATEKOS:"SPC(29)"-----------"
152 PRINTTAB(25)"▒"JN$(2)"  ";:POKE646,BSZ(2)-1:PRINT"▒O▒"
154 PRINTTAB(25)"▒LEPESEK SZAMA:"SPC(72);LSZ(2)
156 RETURN
180 PRINTA$(1);"IGY NEM LEHET                ":PRINT"▒LEPES:";:RETURN
200 L=ASC(M1$)-64
205 IFF(L)=0THENPRINTA$(1);"OTT NINCS BABU!         ":GOTO220
210 IFF(L)=-BSZ(J)+3THENPRINTA$(1);"A BABU AZ ELLENFELEDE!":GOTO220
215 IFF(L)=BSZ(J)THEN225
220 PRINT"▒LEPES:";:GOTO590
225 K=ASC(M2$)-64
227 IFL=1ANDK<>2ANDK<>6ANDK<>7THEN290
228 IFL=2ORL=3ORL=4THEN240
229 IFL=5ANDK<>4ANDK<>9ANDK<>10THEN290
230 IFL=6ORL=11ORL=16THEN250
231 IFL=10ORL=15ORL=20THEN260
232 IFL=21ANDK<>16ANDK<>17ANDK<>22THEN290
233 IFL=22ORL=23ORL=24THEN270
234 IFL=25ANDK<>19ANDK<>20ANDK<>24THEN290
235 GOTO280
237 IFL=10ORL=15ORL=20THEN260
240 IFK=L-1ORK=L+1ORK=L+4ORK=L+5ORK=L+6THEN280
245 GOTO290
250 IFK<>L-5ANDK<>L-4ANDK<>L+1ANDK<>L+5ANDK<>L+6THEN290
255 GOTO280
260 IFK<>L-6ANDK<>L-5ANDK<>L-1ANDK<>L+4ANDK<>L+5THEN290
265 GOTO280
270 IFK<>L-6ANDK<>L-5ANDK<>L-4ANDK<>L-1ANDK<>L+1THEN290
280 IFF(K)=1ORF(K)=2THENPRINTA$(1)"OTT MAR VAN BABU!     ":GOTO220
282 POKEBA(L),81:POKEBA(L)+54272,15:POKEBA(K),215:POKE54272+BA(K),BSZ(J)-1
285 F(K)=BSZ(J):LSZ(J)=LSZ(J)+1:F(L)=0
286 IFJ=1THENGOSUB140:J=-J+3:GOTO620
288 GOSUB150:J=-J+3:GOTO620
290 PRINTA$(1)"IGY NEM LEPHETSZ!     ":PRINT"▒LEPES:";:GOTO590
499 END
500 POKE53280,.:POKE53281,.
505 PRINT"▒ KERED A JATEKSZABALYT? (I/N)";
510 POKE204,0:GETV$:IFV$=""THEN510
515 IFV$="I"THENPOKE204,1:GOSUB20:GOTO530
520 IFV$="N"THENPOKE204,1:GOTO530
525 GOTO510
530 POKE53280,.:POKE53281,.
535 GOSUB60:GOSUB100:GOSUB120:POKE53280,4:POKE53281,4
540 GOSUB140:GOSUB150
550 PRINT"▒A▒B▒C▒D▒E":PRINT"▒F▒G▒H▒I▒J"
555 PRINT"▒K▒L▒M▒N▒O":PRINT"▒P▒Q▒R▒S▒T"
560 PRINT"▒U▒V▒W▒X▒Y"
565 A$(1)="▒"
570 A$(2)="NE HULYESKEDJ!              "
580 IFBSZ(1)=1ANDJSZ=2THENPRINTA$(1)JN$(1);" TE KEZDESSZ!":PRINT"▒LEPES:";:J=1
587 IFBSZ(1)=2ANDJSZ=2THENPRINTA$(1)JN$(2);" TE KEZDESSZ!":PRINT"▒LEPES:";:J=2
588 GOSUB130
589 REM *** KEZDOLEPES
```

```
590 INPUTC$
595 M1$=MID$(C$,1,1):IFM1$<"A"ORM1$>"Y"THENPRINTA$(1);A$(2)"■":PRINT"LEPES:";:GOTO590
600 M2$=MID$(C$,2,1):IFM2$<"A"ORM2$>"Y"THENPRINTA$(1);A$(2)"■":GOTO590
603 IFM2$<"A"ORM2$>"Y"THENPRINTA$(1);A$(2):PRINT"■LEPES:";:GOTO590
605 IFM1$=M2$THENPRINTA$(1);A$(2):PRINT"■LEPES:";:GOTO590
610 GOTO200
620 O=0:FORV=1TO7:IFF(V)=-BSZ(1)+3THENO=O+1:NEXT
625 IFF(10)=-BSZ(1)+3THENO=O+1
630 M=0:FORV=20TO25:IFF(V)=-BSZ(2)+3THENM=M+1:NEXT
635 IFF(16)=-BSZ(2)+3THENM=M+1
640 IFO=7ORM=7THEN700
655 PRINTA$(1);JN$(J)" LEP";:FORV=1TO18-LEN(JN$(J)):PRINT" ";:NEXT:PRINT" "
660 PRINT"■LEPES:";:GOTO590
700 PRINTA$(1)"■";JN$(-J+3);" NYERT!":FORV=1TO17-LEN(JN$(J)):PRINT" ";:NEXT
705 GETV$:IFV$<>""THEN705
710 POKE53280,.:POKE53281,.:PRINT"■■■"
715 PRINTTAB(14)"GRATULALOK!"
720 PRINT"■"JN$(-J+3):PRINT"■ TE NYERTED MEG A JATEKOT"
725 PRINT"■"JN$(J)" ELLEN."
730 PRINT"■■■UJ JATEK? (I/N) I";:INPUT"■■■";V$:IFV$<>"I"THENEND
735 RUN
```

Fogócska

Egy éve vagyok az OCE tagja és kb. másfél éve foglalkozom a C64 programozásával. A pályázatra készített programjaimat sajnos csak kazettán tudom beküldeni, mert floppy nem áll rendelkezésemre.

A ,,FOGÓCSKA'' című programhoz több szerencse vagy reflex kell, mint logika. Ha a keret közelében tartózkodunk és a dobókockán nagyobb szám van, mint amennyit a keretig léphetnénk, a játékos a helyén marad és a lépés joga az ellenfélre száll. A további játékszabályokat a program ismerteti. A vázlatos felépítése a következő:

10—390	Bevezetőkép és játékszabály kiíratása.
400—470	Választás.
480—760	Pálya kirajzolása.
780—920	Az első játékos lépése.
930—1230	,,Gyengébb'' gép lépése.
1240—1750	Erős gép lépése.
1760—1810	Játékosra vonatkozó kiíratások (ki lép stb.).
1820—1950	Hanghatások.
1960—2030	Kiértékelés.
2040—2090	Eredményhirdetés.

Tisztelettel:

Méhész Imre

```
10 gosub 1820:print "■■■■■■":poke 53281,0:poke53280,0
20 print "■          * *   f-o-g-o-cs-k-a   * *      "
30 x=55582:y=55582+27:z=27
40 poke x,1:poke x-1,0:x=x+1
50 if x=55582+z then x=55582: goto 70
60 goto 40
70 pokey,1
80 y=y-1:z=z-1
90 if z=0 then z=27:goto 120
100 if y=55582 then y=55582 +25
110 goto 40
120 gosub 1820: print "■■■■"
130 print "■         keszitette : mehesz imre"
140 a=55823:b=55823+25:c=25
150 poke a,1:pokea-1,0:a=a+1
160 if a=55823+c then a=55823:goto 180
170 goto 150
180 poke b,1
190 b=b-1:c=c-1
200 if c=0 then c=25:goto 220
210 goto 150
220 poke 1283,39: poke 55554,1
```

Játék-pályázat

```
230 poke 1513,39: poke 55785,1
240 poke 1526,39: poke 55798,1
250 poke 1528,39: poke 55800,1
260 gosub 1820:for k=0to1000:next:gosub 1870
270 poke53280,0:poke53281,0
280 printchr$(14):print"J A T E K S Z A B A L Y"
290 print"Ketszemelyes tablajatek. A jatek le-"
300 print"nyege: az ellenfel babujanak elfogasa."
310 print"A lepesszamot a dobokocka ertekevel 1-4"
320 print"hatarozzuk meg, az iranyitogombok meg-"
330 print"nyomasaval.Aki a 10-et eleri, az nyert.IRANYITAS:"
340 print"1 JATEKOS   -   2 JATEKOS"
350 print"Q Fel-@ Fel"
360 print"Z Le-/ Le"
370 print"A Balra-: Balra"
380 print"S Jobbra-; Jobbra"
390 geta$:ifa$=""then390
400 printchr$(142)
410 print ""
420 poke 53280,0:poke 53281,0:print""
430 print " kivel akarsz jatszani ?"
440 print "1 - egymas ellen jatszotok"
450 print "2 - gyenge gep ellen jatszol"
460 print "3 - eroske gep ellen jatszol"
470 input "";t:ift<lort>3then410
480 print ""
490 print "  ┌─┬─┬─┬─┬─┬─┬─┬─┬─┬─┐"
500 print "  │ │ │ │ │ │ │ │ │ │ │"
510 print "  ├─┼─┼─┼─┼─┼─┼─┼─┼─┼─┤"
520 print "  │ │ │ │ │ │ │ │ │ │ │"
530 print "  ├─┼─┼─┼─┼─┼─┼─┼─┼─┼─┤"
540 print "  │ │ │ │ │ │ │ │ │ │ │"
550 print "  ├─┼─┼─┼─┼─┼─┼─┼─┼─┼─┤"
560 print "  │ │ │ │ │ │ │ │ │ │ │"
570 print "  ├─┼─┼─┼─┼─┼─┼─┼─┼─┼─┤"
580 print "  │ │ │ │ │ │ │ │ │ │ │"
590 print "  ├─┼─┼─┼─┼─┼─┼─┼─┼─┼─┤"
600 print "  │ │ │ │ │ │ │ │ │ │ │"
610 print "  ├─┼─┼─┼─┼─┼─┼─┼─┼─┼─┤"
620 print "  │ │ │ │ │ │ │ │ │ │ │"
630 print "  ├─┼─┼─┼─┼─┼─┼─┼─┼─┼─┤"
640 print "  │ │ │ │ │ │ │ │ │ │ │"
650 print "  ├─┼─┼─┼─┼─┼─┼─┼─┼─┼─┤"
660 print "  │ │ │ │ │ │ │ │ │ │ │"
670 print "  ├─┼─┼─┼─┼─┼─┼─┼─┼─┼─┤"
680 print "  │ │ │ │ │ │ │ │ │ │ │"
690 print "  └─┴─┴─┴─┴─┴─┴─┴─┴─┴─┘"
700 p=0:pp=0
710 x=3:y=4:o(o)=81:o(1)=32
720 a=y*40+x+1024-41:pokea,o(o)
730 n=21:m=22:b(o)=160:b(1)=32
740 v=m*40+n+1024-41:pokev,b(o)
750 a=y*40+x+1024-41
760 pokea,o(o):if a=v then 1920
780 a$="labda":gosub 1740:get t$
790 if t$="s" then x=x+r: goto 840
800 if t$="a" then x=x-r: goto 840
810 if t$="q" then y=y-r: goto 840
820 if t$="z" then y=y+r: goto 840
830 goto 780
840 gosub 1820
850 if x<3 then x=x+r:goto 900
860 if x>21 then x=x-r:goto 900
870 if y<4 then y=y+r:goto 900
880 if y>22 then y=y-r:goto 900
890 poke a,o(1)
900 a=y*40+x+1024-41
910 pokea,o(o):if a=v then 1920
920 on t goto 930,1080,1240
930 a$="kocka":gosub 1740:get t$
940 if t$=";" then n=n+r: goto 990
950 if t$=":" then n=n-r: goto 990
960 if t$="@" then m=m-r: goto 990
970 if t$="/" then m=m+r: goto 990
980 goto 930
990 gosub 1870
1000 if n<3 then n=n+r:goto 1050
1010 if n>21 then n=n-r:goto 1050
1020 if m<4 then m=m+r:goto 1050
1030 if m>22 then m=m-r:goto 1050
1040 poke v,b(1)
1050 v=m*40+n+1024-41
1060 pokev,b(o):if v=a then 1980
1070 goto 780
1080 a$="kocka":gosub 1740
1090 if x>n then n=n+r: goto 1150
1100 if x<n then n=n-r: goto 1150
1110 if x=n and y<m then m=m-r: goto 1150
1120 if x=n and y>m then m=m+r: goto 1150
1130 if y=m and x<n then n=n-r: goto 1150
1140 if y=m and x>n then n=n+r: goto 1150
1150 gosub 1870
1160 if n<3 then n=n+r:goto 1210
1170 if n>21 then n=n-r:goto 1210
1180 if m<4 then m=m+r:goto 1210
1190 if m>22 then m=m-r:goto 1210
1200 poke v,b(1)
1210 v=m*40+n+1024-41
1220 pokev,b(o):if v=a then 1980
1230 goto 780
1240 a$="kocka":gosub 1740
1250 if x>n then k=x-n: goto 1410
1260 if y<m then g=m-y: goto 1450
1270 if x<n then l=n-x: goto 1430
1280 if y>m then f=y-m: goto 1470
```

```
1290 if x=n and y<m then d=m-y:goto 1330
1300 if x=n and y>m then c=y-m:goto 1350
1310 if y=m and x<n then h=n-x: goto 1370
1320 if y=m and x>n then i=x-n: goto 1390
1330 if d<=8 then 1490
1340 m=m-r:goto 1650
1350 if c<=8 then 1510
1360 m=m+r:goto 1650
1370 if h<=8 then 1530
1380 n=n-r:goto 1650
1390 if i<=8 then 1550
1400 n=n+r:goto 1650
1410 if k<=8 then 1570
1420 n=n+r:goto 1650
1430 if l<=8 then 1590
1440 n=n-r:goto 1650
1450 if g<=8 then 1610
1460 m=m-r:goto 1650
1470 if f<=8 then 1630
1480 m=m+r:goto 1650
1490 if r=d then 1340
1500 goto 1240
1510 if r=c then 1360
1520 goto 1240
1530 if r=h then 1380
1540 goto 1240
1550 if r=i then 1400
1560 goto 1240
1570 if r=k then 1420
1580 goto 1240
1590 if r=l then 1440
1600 goto 1240
1610 if r=g then 1460
1620 goto 1240
1630 if r=f then 1480
1640 goto 1240
1650 if n<3 then n=n+r:goto 1710
1660 if n>21 then n=n-r:goto 1710
1670 if m<4 then m=m+r:goto 1710
1680 if m>22 then m=m-r:goto 1710
1690 gosub 1870
1700 poke v,b(0)
1710 v=m*40+n+1024-41
1720 pokev,b(o):if v=a then 1980
1730 goto 780
1740 r=int(8*rnd(1))+1
1750 if r=1 or r=3 or r=5 or r=7 then1740
1760 print "dobokocka ";r/2
1770 print "te jossz ";a$
1780 print "az allas "
1790 print "labda ";p
1800 print "kocka ";pp
1810 return
1820 poke 54296,15:poke 54277,25:poke 54278,68:poke 54273,55:poke54272,65
1830 poke 54276,17
1840 for hh=1to50:next
1850 poke54276,0:poke54277,0:poke54278,0
1860 return
1870 poke 54296,15:poke 54277,25:poke 54278,68:poke 54273,55:poke54272,65
1880 poke 54276,129
1890 for hh=1to50:next
1900 poke54276,0:poke54277,0:poke54278,0
1910 return
1920 for q=1to5
1930 gosub 1820
1940 gosub 1870
1950 nextq
1960 pokea,o(1):p=p+1:if p=10 then 2060
1970 goto 710
1980 for q=1to5
1990 gosub 1870:gosub 1870:gosub1870
2000 gosub 1820
2010 nextq
2020 pokev,b(1):pp=pp+1:if pp=10 then 2040
2030 goto 710
2040 print "a kocka gyozott";pp" : ";p
2050 goto 2070
2060 print "a labda gyozott";p" : ";pp
2070 input"akarsz meg jatszani(i/n) :";i$
2080 ifi$<>"i"thenprint"":end
2090 goto420
```

Tisztelt Szerkesztőség!

Ismét egy sokak által hasznosítható különlegességet küldök! Nemcsak azok figyelmébe ajánlom, akiket a CITIZEN120-D programozása érdekel ékezetes betűkkel, s azt a gyakorlatban is használni szeretnék!

A program ötletét a 48000-es sortól kezdődő formázó rutin adta, amely A BASIC programozás magasiskolája C 64-esen című (Novotrade) kiadványban (116. o.) található. A könyvbeli rutin természetesen több helyen is hibás. Akinek van ilyen könyve megkeresheti! A rutin sorszámozását ezért nem változtattam meg. Ez a rutin természetesen a képernyőre írásnál is használható, illetve nagyobb gépekre is átírható; megérdemli a figyelmet!

Ennek a rutinnak egy különleges használatát mutatom be: a bevitt adatokkal egy táblázatot készítek (címmel, fejrésszel, befejezéssel). A szemléletesség érdekében a programlistában a vonalakat nem füzérekből állítottam össze. Cégjelzéssel, címváltoztatással, számmal, dátummal, aláírással akár okmány is készülhet!

A programlistában szereplő értelmetlennek tűnő karakterek a képernyőn és a nyomtatón ékezetes betűk lesznek.

A 49000-es sortól egy részleges és azonos karakterkészlet programozása található képernyőre (nb/gr) valamint 120-D-re! A printer programozása minden mód használatát és betűtípus kinyomtatását úgy biztosítja, mintha eredeti karaktereit használná! Azok tudják ezt igazán értékelni, akiket idegesít az EASY SCRIPT és egyéb program karakter készleteinek furcsa írása: a printer előbb felírja a szó ékezeteit majd alá a szót. Tudni kell azonban, hogy az EPSON eredeti karakterkészletének átprogramozható részét is csak korlátozva használhatjuk! Ezt a programrészt a saját készítésű karakterszerkesztővel készítettem. A képernyőn a RVS karakterek eredetiek maradnak (nem véletlenül), ezt használom ki a tájékoztató kiírásánál!

A táblázatot itt az INPUT segítségével bevitt adatok alapján szerkesztjük meg, de a bevitel helyett a READ/DATA megoldás is használható. Ekkor az alapadatokat DATA-kban adjuk meg, a számíthatókat kiszámítjuk, ezután hívjuk meg a formázó rutint adatonként, a sorokat külön kinyomtatva. Használhatjuk a rutint a SEQ és REL filék adatainak nyomtatásánál is!

A formázó rutin működése elég bonyolult, nem is a leggyorsabb, de tökéletesek a szolgáltatásai:

- a szöveget a megadott helyen bal oldalon kezdi;
- a számok írását tizedesponthoz, illetve jobb oldalra igazítja;
- a felesleges részeket kitölti szóközzel;
- ha kell, a tizedespont után 0-kal egészíti ki a számot.

A kinyomtatandó füzért PF$-ban gyűjti. A programban ez a füzér kiegészül még a táblázat grafikájával is.

Természetesen a szöveg és szám helyét nekünk kell megadni FO$-ban. A 280-as és a 670-es jól szemlélteti mindezt:

- AAAAAAA szerepel a szöveg;
- 99999 az egész számok;
- 999.999 a nem egészek helyén.

```
100 REM * MESSSOFT:EKEZETES TABLAZATIRO
110 REM * FORMAZOTT ADATKIIRAS (PELDA)
120 REM * C64 & CITIZEN120-D
125 REM * 'BASIC PROGRAMOZAS MAGAS-
130 REM * ISKOLAJA C 64-ESEN' 116.O.
140 REM ******************************
150 :
160 POKE53280,6:POKE53281,6:PRINTCHR$(142)"▮▮▮ FELKE▮▮,▮SZU▮▮─▮LO▮▮─▮K..."
170 :
180 REM *** BEALLITAS - VEZERLES ***
190 :
200 CLR:FORI=.TO18:WW$=WW$+"─┬":NEXT:GOSUB49020:REM BETUK
210 PRINT"▮▮ FEJL#CET NYOMTATOM...":GOSUB49520:GOSUB49650:REM CIMBETUK
220 PRINT#1,SPC(6)"#RT#KKIMUTAT!S"
230 PRINT#1,CHR$(27)"1";CHR$(27)"U1";:GOSUB49690
240 PRINT#1," ┌────┬──────┬──────────────────┬──────────┬────────────┐"
250 PRINT#1," IFSZ. IMENNY.  I CIKK MEGNEVEZ#SE  I EGYS.!R    I #RT#K       I"
260 P0$=" ├────┼──────┼──────────────────┼──────────┼────────────┤"
270 P1$=" I    I      I                  I          I            I"
280 PO=.:FO$="999 99999 AAAAAAAAAAAAAAA 999999.99 9999999999.99"
290 PRINT#1,P0$
300 PRINT#1,P0$
310 PRINT#1,P1$
320 PO=PO+1:PRINT"▮MESSSOFT: FORM!ZOTT ADATKI*R!S NYOMTAT$N"
330 PRINT"  I#RT#KKIMUTAT!S I"
340 PRINT"  └─────────────────────────────────┘":PRINT
342 PRINT"BET)K:▮";:FORI=32TO42:PRINTCHR$(I);:NEXTI:PRINT
344 PRINTTAB(6);:FORI=32TO42:PRINTCHR$(I);:NEXTI:PRINT:PRINT
346 PRINT,,"▮▮BEVITEL SZ!MA:"PO
348 PRINT:PRINT:PRINTWW$
350 POKE214,20:PRINT:PRINTWW$
355 IFMJ$<>""THENPRINTMJ$;AN*EP:MJ$=""
360 POKE214,22:PRINT:PRINT" K#REM AZ ADATOKAT [R] ";
370 POKE214,10:PRINT:PRINTTAB(18)"MAX.▮"MID$(FO$,5,4)
380 INPUT"MENNYIS#G <DB/M>[4]:";AN
390 IFAN>9999THENPRINT"▮";:GOTO380
400 PRINTTAB(18)"MAX.▮"MID$(FO$,11,15)
410 INPUT"   MEGNEVEZ#SE [15]:";AR$
420 IFLEN(AR$)>15THENPRINT"▮";:GOTO410
430 PRINTTAB(18)"MAX.▮"MID$(FO$,29,8)
440 INPUT" EGYS#G!R <FT>[5.2]:";EP
450 IFEP>99999.99THENPRINT"▮";:GOTO440
460 IFAN*EP<255*255*125THEN475
470 PO=PO-1:MJ$=" EZ NAGY VOLT :":GOTO320
475 POKE781,22:SYS59903
480 POKE214,22:PRINT:PRINT" J$K AZ ADATOK <I/N> ? ";
490 GETA$:IFA$<>"I"ANDA$<>"N"THEN490
500 IFA$="N"THEN360
510 POKE214,22:PRINT:PRINT" RENDBEN, FELDOLGOZOM. ";
520 FI$=STR$(PO):GOSUB48020:FP$=FP$+CHR$(27)+CHR$(8)+". I "
530 FI$=STR$(AN):GOSUB48020:FP$=FP$+" I "
```

```
540 FI$=AR$:GOSUB48020:FP$=FP$+"| "
550 FI$=STR$(EP):GOSUB48020:FP$=FP$+"| "
560 FI$=STR$(AN*EP):GOSUB48020:FP$=FP$+" |"
570 FP$="|"+FP$
580 PRINT#1,FP$:PRINT#1,P1$:FZ%=.
590 OS=OS+AN*EP
600 IFOS>255*255*125THENPOKE214,21:PRINT:PRINT" NAGY EREDM#NY..V#GE...":GOTO640
610 POKE214,22:PRINT:PRINT" M#G EGY BEVITEL<I/N> ?  "
620 GETA$:IFA$<>"I"ANDA$<>"N"THEN620
630 IFA$="I"THEN320
640 POKE214,22:PRINT:PRINT" %SSZES*TEK, BEFEJEZEM...";
650 PRINT#1,"├──────────┴──────────┴──────────┴──────────┤"
660 PRINT#1,"├───────────────────────────────────────────┤"
670 FO$="            AAAAAAAAAAAAAAAAAAAAA          999999999.99"
680 FI$="% S S Z E S E N [FT]:":GOSUB48020
690 FI$=STR$(OS):GOSUB48020
700 PRINT#1,"|";FP$;"|"
710 PRINT#1,"└───────────────────────────────────────────┘"
720 PRINT#1,CHR$(27)"@":CLOSE1:END
47998 REM ===========================
47999 :
48000 REM *** FORMAZO RUTIN ***
48001 :
48020 IFFZ%=.THENFZ%=1:FL%=LEN(FO$):FP$=""
48040 FI%=LEN(FI$):FB$="":FC%=1
48050 IFMID$(FO$,FZ%,1)=" "THENFZ%=FZ%+1:FB$=FB$+" ":GOTO48050
48060 FM$=MID$(FO$,FZ%,1)
48080 FV%=.:FH%=.:FA%=.:FB%=.:FF%=.
48100 IFMID$(FO$,FZ%,1)="."THENFZ%=FZ%+1:FF%=2:GOTO48180
48120 IFFZ%>FL%THENFA%=FA%+1:FZ%=FZ%+1:GOTO48240
48140 IFMID$(FO$,FZ%,1)=" "THEN48240
48160 FA%=FA%+1:FZ%=FZ%+1:GOTO48100
48180 IFMID$(FO$,FZ%,1)=" "ORFZ%>FL%THEN48240
48200 FB%=FB%+1:FZ%=FZ%+1
48220 GOTO48180
48240 IFMID$(FI$,FC%,1)="."THENFC%=FC%+1:FF%=FF%+4:GOTO48320
48260 FV%=FV%+1:FC%=FC%+1
48280 IFFC%>FI%THEN48380
48300 GOTO48240
48320 IFFC%>FI%THEN48380
48340 FH%=FH%+1:FC%=FC%+1
48360 GOTO48320
48380 IFFM$="A"THEN48780
48400 L2=FA%-FV%:IFL2<.THENFF%=FF%+1:GOTO48720
48420 IFL2=.THEN48460
48440 FORL1=1TOL2:FB$=FB$+" ":NEXTL1
48460 IFFV%=.THEN48520
48480 L2=FV%:FORL1=1TOL2
48500 FB$=FB$+MID$(FI$,L1,1):NEXTL1
48520 IFFF%=.THEN48920
48540 FF%=.:FB$=FB$+".":L2=FV%+2
48560 IFFB%=.THEN48920
48580 IFFB%=FH%THENL3=FB%:GOTO48620
48600 L3=-FH%*(FB%>FH%)-FB%*(FH%>FB%)-1
48620 FORL1=L2TOL2+L3
48640 FB$=FB$+MID$(FI$,L1,1):NEXTL1
48660 IFFH%>=FB%THEN48910
48680 L2=FB%-FH%:FORL1=1TOL2
48700 FB$=FB$+"0":NEXTL1:GOTO48910
48720 L2=FA%+FB%:IF(FF%AND2)THENFF%=1:L2=L2+1
48740 FORL1=1TOL2:FB$=FB$+"*":NEXTL1
48760 GOTO48910
48780 IF(FF%AND4)THENFV%=FV%+FH%+1
48800 L2=FV%:FORL1=1TOL2
48820 IFMID$(FI$,L1,1)<>" "THENL3=L1:L1=L2
48840 NEXTL1
48860 L2=-FV%*(FA%>FV%)-FA%*(FV%>FA%)
48870 IFL2=.THENL2=FV%
48880 FORL1=L3TOL3+L2
48900 FB$=FB$+MID$(FI$,L1,1):NEXTL1
48902 IFFV%>FA%THEN48910
48904 L3=FA%-L2:FORL1=1TOL3
48906 FB$=FB$+" ":NEXTL1
48910 IFFZ%>FL%THENFZ%=.:GOTO48960
48920 L2=FZ%:L3=FL%:FORL1=L2TOL3
48930 IFMID$(FO$,L1,1)<>" "THENFZ%=L1:L1=L3:GOTO48950
48940 FB$=FB$+" "
48950 NEXTL1
48960 FP$=FP$+FB$:FF%=(FF%AND1)
48970 RETURN
48980 REM ===========================
48990 :
49000 REM * BETUK KEPERNYORE *
49010 :
49020 FORA=.TO48:READB:POKE12*4096+A,B:NEXT
49030 DATA162,16,169,,141,14,220,169,51,133,1,169,208,160,,132,34,133
49040 DATA35,169,112,132,36,133,37,177,34,145,36,200,208,249,230,35,230
49050 DATA37,202,208,242,169,55,133,1,169,1,141,14,220,96
49060 SYS49152
49070 READA:IFA=-1THEN49090
49080 FORK=.TO7:READB:POKE28672+A*8+K,B:NEXTK:GOTO49070
49090 POKE53272,189:POKE56576,150:POKE648,108:PRINTCHR$(147)CHR$(8)
```

A kivitelhez helyükre kell képzelni a táblázat vonalait is, hiszen a FO$-hoz ezeket hozzáadjuk PF$-ban. Másképpen nem oldhatjuk meg, mert a formázó rutin csak a leírt karaktereket engedi meg FO$-ban. Megfelelő sorrendben kell a rutint meghívni!

240 : táblázat fejrészének nyomtatása;
260—270: többször használatos füzérek meghatározása;
280: folyószám =0, nyomtatási formátum az adatokhoz;
290—310: fejrész alatti sorok nyomtatása;
320: folyószámnövelés, képernyőinfo;
342—344: ékezetes billentyűk és betűk kiírása a képernyőre;
346: bevitelszám, vonalak, túl nagy érték kiírása;
370—460: adatok bevitele;
— tájékoztatás a formátumról:
— javítás rossz adatok esetén;
470: ha nagy volt a bevitt érték (db és egys. ár szorzata) új bevitel: biztosítva a két tizedesjegy kiírását;
480—500: javítás biztosítása;
510: tájékoztatás a 22. sorba;
520—570: a füzér összeállítása adatokkal, vonalakkal;
— CHR$(8) = visszalépés (backspace):
580: füzér kinyomtatása és táblázatrajz;
590: adatösszegzés (OS-ben);
600: ha nagy az összeg, befejezésre;
610: új adatsor bevitele vagy befejezés;
640: tájékoztató;
650—660: táblázatzárás rajza;
670: zárásformátum;
680—690: zárófüzér összeállítása;
700: kinyomtatása;
710: alsószél rajza;
720: nyomtató teljes törlése (mód, ék. kar. stb.), zárás, vége;
48020—48970: formázórutin (bonyolultsága miatt nem részletezem);
49020—49060: karakterkészlet (nb/gr); ⟨C=SHIFT⟩ tiltás; 64 karakterek ROM-ból RAM-ba
49070—49190: RAM-ban karakterváltoztatás (képernyőn láthatók lesznek);
49230—49250: nyomtatásellenőrzés; megnyitás, alapállapot;
49260: printer ROM printer RAM-ba;
49270—49300: karakterek átírása RAM-ban;
49520: printer RAM használata (új karakterekkel);
49530: printer ROM használata, ha eredeti karaktereket akarom használni, (képernyő marad!);
49650: széles betűtípus választás;
49660: kétszeres betűmagasság választás;
49690: betűtípus választás a táblázathoz;
49700: betűmagasság eredeti.

A bevitt adatok után kicsit várni kell, amíg a füzér elkészül, mert csak akkor hívjuk meg a formázórutint, ha az adataink jók, akkor viszont többször. A munka gyorsabbnak tűnik, ha a bevitel után azonnal külön-külön hívjuk meg a formázórutint. Ehhez viszont a javítási technikát is módosítani kell.

Tisztelettel:

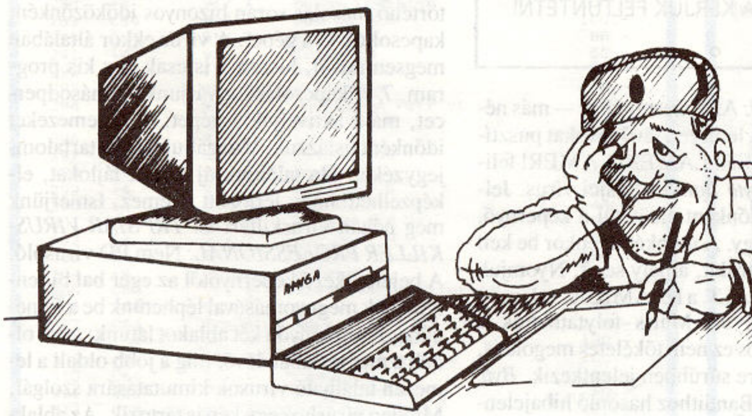
Mesterházi Sándor

```
49100 DATA42,12,8,60,24,24,24,60,
49110 DATA41,54,36,102,102,102,102,60,
49120 DATA40,102,,102,102,102,102,60,
49130 DATA39,12,8,102,102,102,102,60,
49140 DATA38,54,36,60,102,102,102,60,
49150 DATA37,102,,60,102,102,102,60,
49160 DATA36,12,8,60,102,102,102,60,
49170 DATA35,12,8,126,96,124,96,126,
49180 DATA33,12,8,24,60,102,126,102,
49190 DATA-1
49200 :
49210 REM * BETUK PRINTERRE *
49220 :
49230 OPEN1,4,7:CLOSE1:IFST=0THEN49250
49240 PRINT" NYOMTAT$T NEM TAL!LOM.▢":GOTO49230
49250 CLOSE1:OPEN1,4:PRINT#1,CHR$(27)"@";
49260 PRINT#1,CHR$(27)":"CHR$(0)CHR$(0)CHR$(0);
49270 PRINT#1,CHR$(27)"&"CHR$(0)CHR$( 33)CHR$( 33);
49280 FORI=1TO12:READX:PRINT#1,CHR$(X);:NEXTI
49290 PRINT#1,CHR$(27)"&"CHR$(0)CHR$( 35)CHR$( 42);
49300 FORI=1TO96:READX:PRINT#1,CHR$(X);:NEXTI:RETURN
49310 DATA128,14,16,36,,100
49320 DATA128,36,16,14,,:REM!=A,
49330 DATA128,62,,42,,106
49340 DATA128,42,,34,,:REM#=E,
49350 DATA128,28,34,,34,64
49360 DATA162,,34,28,,:REM$=O,
49370 DATA128,28,34,128,34,
49380 DATA34,128,34,28,,:REM%=O..
49390 DATA128,28,34,64,162,
49400 DATA34,64,162,28,,:REM&=O,,
49410 DATA128,60,2,,2,64
49420 DATA130,,2,60,,:REM'=U,
49430 DATA128,60,2,128,2,
49440 DATA2,128,2,60,,:REM(=U..
49450 DATA128,60,2,64,130,
49460 DATA2,64,130,60,,:REM)=U,,
49470 DATA128,,,34,
49480 DATA126,128,34,,,:REM*=I,
49500 :
49510 REM * KARAKTERKESZLET VALASZTAS *
49520 PRINT#1,CHR$(27)"%1"CHR$(0);:RETURN:REM 120D EK.BE
49530 PRINT#1,CHR$(27)"%0"CHR$(0);:RETURN:REM 120D EK.KI
49630 :
49640 REM * CIMFELIRAT *
49650 PRINT#1,CHR$(27)"!"CHR$(48);:REM TIP
49660 PRINT#1,CHR$(27)CHR$(126)"11";:REM MAGAS BE
49670 RETURN
49680 REM * VISSZA NORM *
49690 PRINT#1,CHR$(27)"!"CHR$(17);:REM TIP
49700 PRINT#1,CHR$(27)CHR$(126)"10";:REM MAGAS KI
49710 RETURN

READY.
```

AMIGA ROVAT

FRIEND TWO CREW



Az Amiga Üzembehelyezése, és tartozékai

Mit tegyünk, ha még soha sem láttunk Amigát, vagy nem rég vásároltunk egyet? A gép csomagjában három fő részt találunk. A központi egységet az alapgépet, az igen nehézre sikeredett tápegységet, valamint a mechanikus egeret. A lemezmeghajtó egységet a drive-ot az alkotók a 64-es géppel ellentétben beleépítették az alaplapba. A dobozban találunk még két, vagy több 3,5-ös mágneslemezt, valamint az Amiga-Basic és a DOS kézikönyvet. Helyezzük el a berendezést, és figyeljük meg a hátoldalon elhelyezkedő ki- és bemeneteket. Jobbról balra haladva a következőket találjuk. Az első pólus az egérbemenet, amely joystick (botkormány) bemeneteként is használható. Csomagoljuk ki az egeret és máris csatlakoztassuk. A következő bemenet a központi botkormány csatlakoztatására való. A joystickokat külön kell megvásárolnunk. Kezdő Amigásoknak lapkapcsolós, és mikrokapcsolós botkormányokat ajánlunk. Az Amiga ugyanazt a joy.-t használja, mint a Commodore 64-es. Ezután az RCA kimenet következik. Innen köthetjük hi-fi berendezésünkhöz, rádiómagnónkhoz számítógépünket. Érdemes erre a célra egy négy RCA, öt dines sztereó csatlakozót vásárolni. A kimenetről monitort is csatlakoztathatunk, természetesen audió formájában. A következő ki- és bemenet 23 pólusú, és ide illeszthetjük a lemezmeghajtó (külső) adatátviteli kábeljét. Mellette az ún. serial port található, ez adatközlési célokat szolgál (például nyomtatók esetében (.A következő a parallel port/párhuzamos kimenet), ide lehet kötni a legtöbb nyomtatót. A következő négyzet alakú bemenethez a tápegységet illesztjük. Csomagoljuk ki, és mielőtt a hálózathoz kötnénk, helyezzük be a csatlakozót az Amiga említett bemenetébe. Az utolsó előtti csatlakoztatási pont a monitorkimenet. Ha nincs monitorunk, és televízióhoz akarjuk csatlakoztatni az Amigát ún. *modulátorra* van szükségünk. Szakboltban vásárolható meg. A televíziót a következőképpen köthetjük a frekvenciaátalakító modulátorhoz. Az RF kábelt, antennakábelt, az out jelzésnél először a modulátorba, a másik végét pedig a televízió antennabemenetébe helyezzük. A modulátorhoz egy kisebb, egyik oldalán egy-, a másikon kétpólusú végződéssel ellátott csatlakozó is tartozik. Ha a számítógép hangját is a televízión keresztül akarjuk hallgatni, akkor a kétpólusú részt az előbb említett hi-fi kimenethez kell csatlakoztatnunk, az egypólusú részt pedig az átalakító audió in bemenetéhez. A legutolsó mono feliratú kimenetre is lehet tv-t kötni. Ebben az esetben fekete-fehérben élvezhetjük az Amigát. A különböző portok megismerése után csatlakoztassuk a tápegységet a 220 V-os hálózathoz. Ha ez megtörtént, a számítógépet a tápon található kapcsoló átállításával tudjuk üzembe helyezni. A televízió (monitor) képernyőjén megjelenik egy lemezt tartó kéz. Fontos még megjegyeznünk, hogy az Amiga a játék-, felhasználói, demó programokat automatikusan tölti (DOS-os, trackload-os). A dobozban a kicsomagolás során találtunk két lemezt. Az egyikre rá van írva Workbench (vörkbencs). Ez a lemez tartalmazza az Amiga operációs rendszerének tartozékait. A lemezen található file-okról, a workbench menüjéről, az ikonokról, a könyvtárakról írásunkban nem foglalkozunk. A workbench-ről bővebben az 1990-ben kiadott Nagy Amigakönyv foglalkozik. Megvásárolható minden szakboltban. Mivel már gépünk üzemképes, célszerű olyan cégekhez fordulnunk, melyek játékprogramokat, demókat forgalmaznak.

Hardverkiegészítők

A memóriabővítők az alapgép kapacitását növelik. A legegyszerűbb bővítő a Commodo-

Kedves Olvasónk!

A Commodore cég 1985-ben kifejlesztett Amiga néven forgalomba hozott számítógépeivel foglalkozunk. Azokat a számítógépbarátokat szeretnénk megnyerni, akik a havi lapban megjelenő programlisták felhasználása mellett érdeklődnek gépük más lehetőségei iránt is. A grafikai és zenei képességen túl demókkal, felhasználói programokkal, valamint játékok leírásával foglalkozunk. Lehetőségeinkhez mérten a programok leírása mellé grafikai, képi illusztrációk sokaságával szeretnénk rovatunkat bővíteni. Beszámolunk a nemzetközi software-piacon megjelent újdonságokról, és azok hivatalos árairól. A szerkesztését a BUMBLE BEE nevű csapat fogja elvégezni. Ha kedvet érez az említett témakörön belül programleírással, cikkekkel rovatunkban szerepelni, a következőket tegye. A leírt (gépelt, nyomtatott) szöveget, valamint egy darab bélyeggel ellátott válaszborítékot, címének és nevének pontos feltüntetésével a következő levélcímre küldheti: FRIEND TWO CREW 1399 Bp. Pf.: 701/55. A leírásokat nem tudjuk anyagiakhoz kötni, az elismerés csak erkölcsi lehet. Örömmel veszünk olyan írásokat, cikkeket, melyek más géptípusokkal foglalkoznak. A beérkezett munkákat a szerkesztőséggel megvitatjuk, a közlésre alkalmasakat a soron következő számok valamelyikében rovatunkban megjelentetjük. A következő írásunkat az újdonsült Amiga tulajdonosoknak ajánljuk. Az újság legközelebbi számában a vírusokról, a vírusvédelemről készült cikkünket fogjuk, a bevezetésben felvett témakörök mellett közölni.

EZÚTTAL SZERETNÉNK FELHÍVÁST INTÉZNI OLVASÓINKHOZ! SAJÁT, VAGY CSOPORTOSAN ELKÉSZÍTETT MUNKÁIKAT, LEÍRÁSAIKAT, CIKKEIKET ÖRÖMMEL FOGADJUK, ÉS AZ ARRA ÉRDEMESEKET ROVATUNKBAN KÖZÖLJÜK. REMÉLJÜK, HOGY SOK AMATŐR CIKKÍRÓVAL TALÁLKOZUNK. AZ *AIR SERVICE* REPÜLŐGÉPES SZOLGÁLAT JÓVOLTÁBÓL A BEÉRKEZETT IROMÁNYOK KÖZÖTT BUDAPESTI SÉTAREPÜLÉST SORSOLUNK KI. A GÉPELT, VAGY NYOMTATOTT SZÖVEGET TARTALMAZÓ LEVELET AZ 1399 BP. PF. 701/55 LEVÉLCÍMRE LEHET ELKÜLDENI. (A CÍMZETT ROVATBA KÉRJÜK FELTÜNTETNI FRIEND TWO CREW (CIKK)).

re A501-es kártya. Ez az Amiga 500 alján lévő fedél levétele után a nyáklapon elhelyezkedő portra köthető. A Combitec DRAM bővítők az 500-as RAM-ját 2, 4, 8 Mbyte-tal növelik. A felsorolt bővítők hibája, hogy a tárolt adatok a gép kikapcsolása után megsemmisülnek. A Combitec S-RAM 500/1M típusú 512 kbyte-os bővítője már képes az adatokat több hónapig tárolni. A Combitec dinamikus bővítői az A2000, A3000, A4000-hez is megvásárolhatók. Az A2058 jelzésű bővítő 2, 4, 8 mbyte kapacitású. Ezek a bővítők nagyon drágák, ezért érdemes a párezer forintba kerülő A501-es bővítőt megvásárolnunk. Az A500-ashoz 1993-ban már csaknem 1 MB memóriaigényű programokat készítenek, ezért szükséges a bővítő.

Vírusok, vírusvédelem

A lehető legkülönfélébb módon igyekeznek megakadályozni a különféle programok használatát. Általános jellemzőjük, hogy képesek önmagukat szaporítani. Az ún. IRQ-vírusok leggyakrabban a programok végére fűződnek, és a program belépési címét átírják a saját belépési címükre. Ezeket a cseles kis programokat leggyakrabban unatkozó programozók fabrikálják egymás ugratására, sajnos ennek általában a felhasználók is megisszák a levét. Szerencsére mindig akad olyan programozó is, aki a vírusellenes softverek (víruskillerek) készítésével foglalkozik. A víruskillereket többnyire szabadon terjeszthető programként szokták a felhasználók rendelkezésére bocsátani. A vírusok a fertőzött lemezről bekerülnek a gép memóriájába. A számítógép kikapcsolásáig több tucat lemezt is észrevétlenül megfertőzhetnek. Igen nagy veszélyt jelenthet a merevlemezes kiegészítővel rendelkezők számára. A vírus a gép kapacitását több Mbájttal csökkentheti. Vannak ártalmatlan vírusok is, amelyek gyakran megtréfálhatják a felhasználót. Ilyenkor nem károsodik semmi, csupán időleges zavart tapasztalunk. A felhasználó sokszor nem tudja eldönteni, hogy ártalmatlan vírussal áll-e szemben. Vannak olyan vírusok, amelyek az illetéktelen másolás ellen lettek létrehozva. A vírusfertőzéshez az is elegendő, ha a még nem fertőzött lemezt olyan gépbe tesszük, amelyben előzőleg vírussal fertőzött lemez volt. A vírushatások változatossága kimeríthetetlen, hiszen az emberi leleményesség is határtalan. A vírus miatt nem tudjuk a programot folytatni, amely egy játékprogram esetében csupán bosszantó, de egy szövegszerkesztő vagy egy adatbázis-kezelő működtetésében már gondot okoz, hiszen több órai munkánk veszhet kárba. Ismerkedjünk meg néhány elterjedt Amiga vírussal, és víruskillerrel. *Exterminator:* Az Exterminátor — más néven LAMER — a lemezen lévő sávokat pusztító módon a LAMER! LAMER! LAMER! felirattal írja tele. *Byte Bandit:* Svájci vírus. Jellemzője, hogy időnként elsötétül a képernyő, és a program lefagy. A munkát ilyenkor be kell fejezni. Van egy trükk, amely segít. Nyomjuk le egyszerre a két ALT, a két AMIGA, valamint a SPACE billentyűt. Máris folytathatjuk a programot. Sajnos ez nem tökéletes megoldás, mivel a hiba egyre sűrűbben jelentkezik. *Byte Warrior:* A Byte Bandithoz hasonló hibajelenséget tapasztalunk. Disk monitorral vizsgálva a fertőzött lemezt a DASAO.2 részletet találjuk. Ezért DASA vírusnak is szokták nevezni. *IRQ:* Link vírus. Megkeresi a Startup-sequence második programját, és megpróbálja megfertőzni. Ha ez nem sikerül, akkor hozzákapcsolja magát a DIR parancshoz. Ha írásvédett lemezt teszünk a meghajtóba a Disk xy is write protected please remove felirattal próbálja kikapcsoltatni a védelmet a gép kezelőjével. Ez nem butblokk vírus. *SCA vírus:* Svájci. Kárt a butblokkot használó programokban okoz. Persze az is elég, hogy a számítógép időnként lefagy, és csak reszetelni tudunk. Ha a gépet nem kapcsoljuk ki, csak a memóriát töröljük, a vírus hamarosan bejelentkezik. Monitorral vizsgálva a Something wonderful has happened. Your Amiga is alive feliratot találjuk. A vírus számos fajtáját ismerjük, pl. az *AEK*, az *LSD*, az *OBELISK*. *MECHANIZMUS:* Az Amiga a lemez ismertetőjelét automatikusan olvassa be. A gép megkeresi a DOS jelzést, és ha megtalálja elindítja az alaprutint. A start rutin végrehajtása során kerülhet be a vírus. Módosítja a reszet rutint, és helyet foglal a memóriában. Reszetelés esetén mindig hibajelzést kapunk. Disk monitorral vizsgálva a fertőzött lemezt az első két blokk eltér egy vírusmentes lemez említett blokkjaitól. *Butblokk vírus:* Az Amiga butblokkjában rejtőzködik. Igen rövid program. Felülírja az eredeti programot, és reszetálással sem tűntethető el. *Trójai faló:* Nem a butblokkot használja, ha nem be van épülve a programba. Az illetéktelen másolás ellen készült. A másolaton egy kis részlet nem egyezik meg az eredetivel. Ekkor a vírusprogram aktiválódik és fokozatosan megsemmisíti a sofvert. Nehezen észrevehető, ezért több száz másolat is elkészülhet a hiba észleléséig. Néhány butblokk vírus: GADAFFI, TIME, BOMB, REVENGE, NORTH STAR... stb. *Védekezés.* A betegség elleni védekezés legjobb módja a megelőzés. A következőképpen tudjuk ezt elérni: 1. Azokat a lemezeket, amelyekről csak betöltjük az adatokat, mindig írásvédett állapotban használjuk. 2. Ha nem gyári lemezről másolunk, akkor a végrehajtás előtt ellenőrizzük, hogy nem volt-e jelen vírus. 3. Ha lehet, a vásárolt vagy általunk írt fontosabb programokról még a legelső felhasználás előtt készítsünk biztonsági másolatot, egy vírusmentes másolóval. 4. A merevlemezes tulajdonosok csak az ellenőrzés után aktiválják a winchestert. 5. Néha az segít, ha figyeljük a program betöltésének fázisait, és az esetleges eltéréseknél reszetelünk. 6. Copy-val történő másolás során bizonyos időközönként kapcsoljuk ki a gépet. A vírus ekkor általában megsemmisül, hiszen ő is csak egy kis program. 7. Kikapcsoláskor várjunk tíz másodpercet, majd aktiváljuk a gépet. 8. A lemezeket időnként listázzuk. Vizsgáljuk meg a tartalomjegyzéket. Ha találunk új furcsa fájlokat, elképzelhető hogy fertőzött a lemez. Ismerjünk meg néhán víruskillert is. *TRI STAR VIRUSKILLER PROFESSIONAL:* Nem PD vírusölő. A bejelentkező képernyőtől az egér bal billentyűjének megnyomásával léphetünk be a főmenübe. A képernyőn két ablakot látunk: a bal oldali a memóriában lévő, míg a jobb oldali a lemezen található vírusok kimutatására szolgál. Minden vírushoz egy kép is tartozik. Az ablakban a kép megjelenése jelzi a keresés közben megtalált vírust. Az ismert vírusok: NORTHSTAR?SCA? DASA?DISK DOCTORS?GADAFFI? LAMER. A menüben az alábbi négy parancs közül választhatunk: KILL VIRUS: kiírja a memóriában lévő azonosított vírust. INSTALL DISK: törli a meghajtóban lévő lemez butblokkját és oda egy védett butblokkot másol. PROTECT:SCA és a DASA vírusok ellen védelmet helyez a mágneslemezre. DUMP: a képernyőre listázza a meghajtóban lévő lemez butblokkját ASCII formában. A másik program a *VIRUS EXPERT* néven fut. Képes a BootBlock vírusokat megtanulni, és a későbbiek során felismerni. Sajnos az IRQ vírusok ellen alkalmatlan. *Vírus Checker* napjaink legjobb vírusölője, több mint 69 verziója létezik. Felismeri a következő vírusokat, és megsemmisíti őket: SCA, AEK, LSD, BYTEBANDIT, REVENGE, BYTEWARRIOR, NORTHSTAR, OBELISK, IRQ, PENTAGON, HCS, DISK DOCTORS, LAMER EXTERMINATOR, TIME BOMBS, CCCP, LIBERATOR stb. A Vírus Checkerrel a Commodore újság 93/5-ös számában már kollegánk foglalkozott. Érdemes ezt a számot a szerkesztőség irodájában, vagy postai utánvéttel megvásárolni.

Hardverkiegészítők

MONITOROK: Commodore 1081-es monitor már csak használtan kapható. A legelterjedtebb típus az 1084-es, amelynek Scart, RGB és FBAS bemenete is van. Ennek sztereó változata az 1084 S jelzésű készülék. A felsorolt készülék hibája, hogy Interlace üzemmódban remeg a kép. Ez a jelenség a 2024-es 1008×800 felbontású monitoron már nem érzékelhető. Philips CM 8833 típusú monitor felbontása jóval nagyobb, mint az 1084 S készüléké. Aki PC üzemmódban használja az Amigát, annak érdemes megvásárolni a Multisync II névre hallgató monitort. A piacon még ismert monitorok a Highscteen KP 748, Philips 8833 II. A C-1084 S ára 25—30 ezer forint körül változik.

PÖTYÖGŐ SZOLGÁLAT

282	C64	91/ 5	Raster Split	50.-
283	+4	91/ 6	Programnyilvántartó	80.-
284	C64	91/ 6	Regiszter	60.-
285	C64	91/ 6	Digitalizált zene	50.-
286	C64	91/ 6	Ékezetes MPS 802	80.-
287	C64	91/ 7	*Ékezetes iratkészito	100.-
288	+4	91/ 7	Bűvös téglalap	100.-
289	C64	91/ 7	Dominó - H	100.-
290	C64	91/ 7	Raktáros	80.-
291	+4	91/ 7	Tologató	60.-
292	C64	91/ 7	Aren 2000	50.-
293	C64	91/ 7	Bad Lamps	80.-
294	C64	91/ 7	Operation Patriot	50.-
295	+4	91/ 7	Mozaik - B	90.-
296	+4	91/ 7	Logi Kód	60.-
297	C64	92/ 1	Databevivő	50.-
298	+4	91/ 7	Bomber	50.-
299	+4	91/ 7	Dominó - K	50.-
300	+4	91/ 7	The Wall	40.-
301	C64	91/ 7	Mozaik - K	50.-
302	+4	91/ 7	Dominó - R	60.-
303	+4	91/ 7	*Borgok Kincse	190.-
304	+4	91/ 7	Asteroids	60.-
305	C64	91/ 9	Ékezetes irat forrás	40.-
306	C64	91/ 9	Yugo2	50.-
307	C64	91/ 9	Adventure - Sz	130.-
308	C64	91/ 9	Morze Oktató B.	130.-
309	+4	91/ 9	Memory Game Plus	100.-
310	C64	91/ 9	*Capitaly	130.-
311	C64	91/ 9	*Gengszterek	60.-
312	C64	91/ 9	*Küldetés 2000	100.-
313	C64	91/ 9	Itt a $ hol a $	50.-
314	C64	91/ 9	Hewing	50.-
315	+4	91/10	Mikrobi	50.-
316	+4	91/10	Nest of Fleas	190.-
317	+4	91/10	Adventure - F	100.-
318	+4	91/10	A világ szeme	150.-
319	C64	91/10	Gondolkozó	100.-
320	C64	91/10	Memori - B	130.-
321	+4	91/10	Matekastély	100.-
322	C64	91/11	Lottó	80.-
323	C64	91/11	Totó	80.-
324	C64	91/11	Német gyakorló	60.-
325	+4	91/11	Beszélő szótár	130.-
326	+4	91/11	File examiner	60.-
327	C64	91/11	Sprite szerkesztő	50.-
328	C64	91/11	Egyszínű Sprite	50.-
329	C64	91/12	Rulett javítás	50.-
330	C64	91/12	Tőzsde	130.-
331	C64	91/12	IQ teszt	150.-
332	C64	92/ 1	Kalandlap	80.-
333	C64	92/ 1	Levélíró	80.-
334	C64	92/ 1	Képernyőmaszkoló	130.-
335	C64	92/ 1	Sprite Kezelő	100.-
336	C64	92/ 1	SI - OPART	40.-
337	C64	92/ 2	Patience Quartet	130.-
338	C64	92/ 2	Monopoly	130.-
339	C64	92/ 2	Akasztófa	80.-
340	C64	92/ 2	Karaktertervező	60.-
341	C64	92/ 3	String Array Manager	100.-
342	C64	92/ 3	Sprite Basic	50.-
343	C64	92/ 3	Screen Help	50.-
344	C64	92/ 3	Gyors validate	60.-
345	C64	92/ 3	BASIC SYNTAXER I.	190.-
346	C64	92/ 4	Tologató C64-re	100.-
347	C64	92/ 4	ABC rendező	100.-
348	C64	92/ 5	*Matek + Római számok	100.-
349	+4	92/ 6	Tesztkészítő + értékelő	100.-
350	C64	92/ 3	BASIC SYNTAXER II.	190.-
351	+4	92/ 7	Három játék	110.-
352	+4	92/ 7	Szökés	80.-
353	C64	92/ 7	Toto	150.-
354	C64	92/ 7	DIGIT analízis program	110.-
355	+4	92/ 7	SCREEN DUMP	150.-
356	C64	92/ 9	UNISEQ filekezelő	130.-
357	C64	92/ 9	Graphic Copy	60.-
358	C64	92/10	Printfox Resizer	130.-
359	C+4	92/11	Tologató II.	150.-
360	C+4	92/12	Sprite Plus-4-en	130.-
361	C64	92/12	Változólista	150.-
362	C64,+4	93/ 2	Szomszéd játék	200.-
363	C64	93/ 2	Táblás játék	200.-
364	C64	93/ 2	SEDA	150.-
365	C64	93/ 2	415 * 200	120.-
366	+4	93/ 2	OTP hitel	100.-
367	C64	93/ 3	Catalog-Check(CTC)	150.-
368	C64	93/ 3	Pattogó golyók	200.-
369	C64	93/ 3	Beszorítós (DO-GUTI)	150.-
370	C64	93/ 3	*Zsírozás	200.-
371	C64	93/ 4	Attrib 64	50.-
372	C64	93/ 4	Táblázatnyomtató	80.-
373	C+4	93/ 4	Borítékkészítő	80.-
374	C+4	93/ 4	TV-óra	50.-
375	C64	93/ 4	Mini Domino	150.-
376	C64	93/ 4	Megathello	120.-
377	C64	93/ 4	Reversi	120.-
378	C64	93/ 4	Tengeri csata	120.-
379	C64	93/ 5	Videokazetták	80.-
380	C+4	93/ 5	DIN-karakterek	80.-
381	C64	93/ 5	DIRYDIRY	80.-
382	C64	93/ 5	Birodalom	150.-

A *-gal jelölt programok csak lemezen rendelhetők !

Kérem, hogy a **Commodore Újság** legközelebbi számában jelenjen meg a következő szövegű apróhirdetés:

(minden kockába egy betűt írjon)

Tagsági szám: ....

Feladandó az újság címére:
Commodore Újság
Budapest, 1388. Pf. 86.

Alulírott megrendelem a következő programokat a PÖTYÖGŐSZOLGÁLAT-tól:

PROGRAM SORSZÁMA ÁRA

1. ,– Ft
2. ,– Ft
3. ,– Ft
4. ,– Ft
5. ,– Ft
6. ,– Ft
7. ,– Ft
8. ,– Ft
9. ,– Ft

Összesen: db ,– Ft

A programokat a SZOLGÁLAT által biztosított
lemezre (99 Ft/db)
kazettára (40 Ft/db)
az általam küldött adathordozóra kérem.
(Kérjük a megfelelő szöveg aláhúzását!)

Postaköltség + munkadíj

A fizetés módja: személyesen — csekken — utánvéttel
(Kérjük a megfelelő szöveg aláhúzását!)

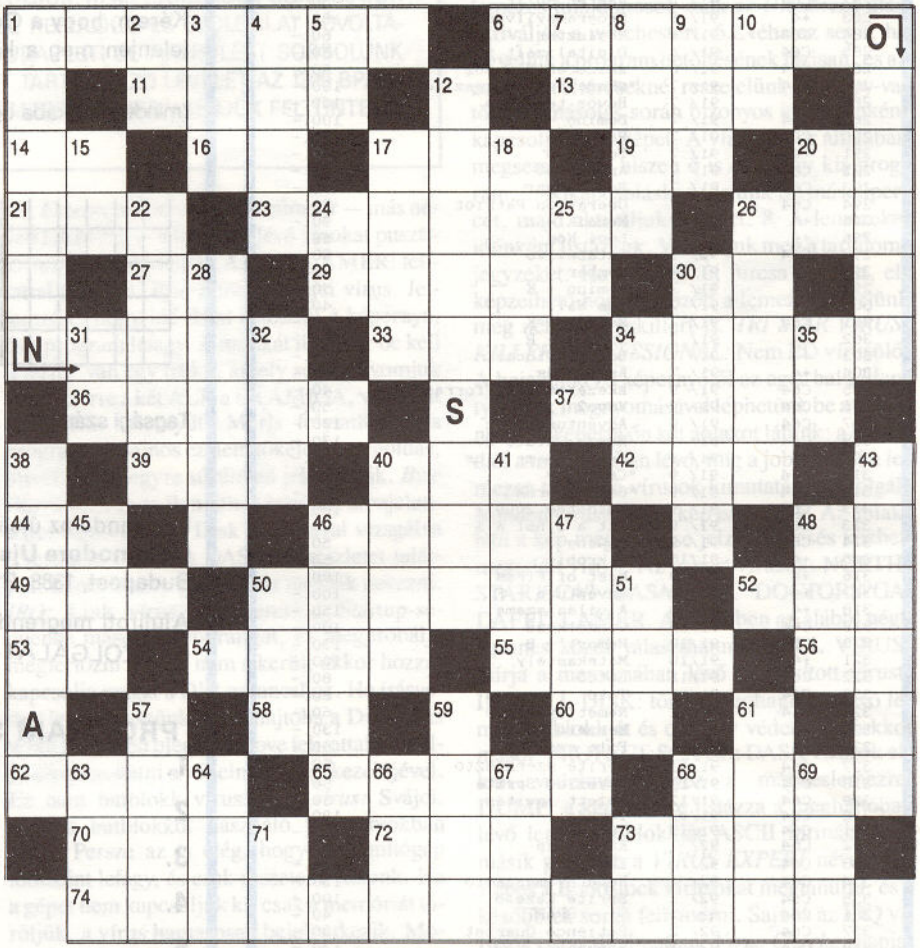
ÖSSZESEN:

A megrendelő neve: ....

Címe: ....

BOLYAI JÁNOS (1802–1860)

KERESZTREJTVÉNY



A meghatározásokat nem választottuk külön, hanem a számozás sorrendjében folyamatosan adjuk. Ha egy számtól mindkét irányba indul ki a megfejtés, akkor előbb a vízszintes, majd egy gondolatjellel elválasztva a függőleges meghatározás szerepel.

MEGHATÁROZÁSOK: 1. Markó Pál költő egyik versének címe. — **Ennyi éves volt Bolyai János mikor meghalt.** 2. Kambodzsai és kubai gépkocsik jelzései. 3. Veszprém megyei helység. 4. Bejgli fajta. 5. Személyesnévmás többese. **6. Bolyai János feleségének leánykori neve.** 7. Botswanai gépkocsi jelzése. 8. Pénzintézet. 9. Értéke. 10. Nikkel vegyjele. 11. Sáv. **12. Itt született Bolyai János matematikus.** 13. Város Dél-Olaszországban. 14. Római számok. 15. Korrövidítés. 16. Két magánhangzó. 17. Nyakbavaló. — Jugoszláviai városból való. 18. Szülő anyák védőszentje. 19. Nátrium vegyjele. 20. Tetejére. — Azonos a vízszintessel. 21. Sportol. 22. Becézett leánynév. 23. Itt született Bayer Ilona tv-bemondónő is. 24. Szív közepe. 25. Állóvíz fordítva. 26. Heves megyei helység. — Van ilyen kereső is. 27. Ételízesítő. 28. Nép. 29. Ceruza. 30. Irányszó. — Evőeszköz. 31. Y. S. 32. Króm és jód vegyjele. 33. Folyó. 34. Magános település. — Kacsát etet. 35. Azonos a 31. függőlegessel. 36. Félretett kártya. 37. Villásreggeli. **38. Utópisztikus társadalomtudományi műve volt Bolyai Jánosnak.** 39. Névelős sporteszköz. 40. Étkező. — Ilyen tanulmány is van. 41. Nem minket. 42. Nem az. 43. Markó Pál másik versének címe. 44. Napszak mássalhangzói. 45. Van szeme. 46. Fejfedő. — Igen szegény. 47. Latin citátum. 48. Ezüst vegyjele. — Vonatkozó névmás. 49. Dunamenti város. 50. Virágtartója. — Hal fajta. 51. Heves megyei helység. 52. Rendben. 53. Ketté(!). 54. Férfinevet. 55. Lóverseny szó. 56. Betűt vet. 57. Ilyen vetés is van. 58. Törökországi gépkocsi jelzés. 59. Szerszám. 60. Kén és kálium vegyjele. 61. Vissza: férfinév. 62. Idegen férfinév. 63. Elődöt. 64. Vissza: napszak. 65. Azonos a 28. függőlegessel. 60. Oly keveredés. 67. Saller egyik fele. 68. Az elektromos feszültség gyakorlati egysége. — Öv. 69. Titokban figyel. 70. Kikötői hajóhíd. 71. Gönc egyik fele. 72. Időhatározó. 73. Emberiség leghőbb vágya. **74. Ebben a betegségben halt meg Bolyai János matematikus.**

A rejtvényt nem kell beküldeni, mert szórakoztató jellegű!

Mokos István

Egyszerű betöltés

Eléggé közismert, hogyan lehet a programokat egyszerűen betölteni: Behívjuk a directoryt, a kurzorral a kívánt programnévre lépünk, beadjuk a DLOAD utasítást, majd az ⟨ESC⟩ ⟨K⟩ funkcióval a sor végére lépünk, s ott a ⟨DEL⟩ háromszori megnyomásával kitöröljük a fölösleges betűket. A ⟨RETURN⟩ megnyomására a kívánt program máris betöltődik. Más lehetőség a DLOAD és a ⟨TAB⟩ háromszori használata. Ezután egy kettőspontot kell kitennünk. A ⟨RETURN⟩-re itt is betöltődik a tartalomjegyzékből kiválasztott program.

De mire valók akkor a funkciós billentyűk? Az ⟨F2⟩-es ezek közül már eleve föl van szerelve a DLOAD " karakterekkel. Akkor pedig érdemes ezt a gombot kissé átdefiniálni:

KEY 2," DLOAD"+CHR$(9)+CHR$(9)+CHTR$(9)+":"
+CHR$(13)

Ha ezek után nyomjuk meg az ⟨F2⟩-es gombot, a legutóbbi magyarázat szerint hívjuk be a kiválasztott programot. Ha azt a behívás után azonnal el akarjuk indítani, vagy a ⟨F2⟩-t, vagy az ⟨F6⟩-ot definiáljuk át:

KEY 6," RUN"+CHR$(9)+CHR$(9)+CHTR$(9)+":"
+CHR$(13)

Hogy az adott funkciós billentyű foglaltságot állandóan használhassuk, érdemes azt egy programba írni, s az egészet egy lemez első állományaként kezelni. Ekkor ugyanis a ⟨SHIFT⟩ és a ⟨RUN/STOP⟩ billentyű megnyomásával azonnal el lehet indítani. A kívánt funkciós billentyű foglaltságok így villámgyorsan (és gombnyomásra) a rendelkezésre állnak.

Véletlen számok

Véletlen számokat jól lehet asszemblerben egy rekeszben létrehozni. Erre a legcélszerűbb a SID-et alkalmazni. A SID harmadik hangját susogásra kapcsolni (129-es hullámforma). Ekkor véletlen zörejek keltődnek hardver oldalról. Ezeket le lehet kérdezni és ezek nagyon jó és mindenekelőtt valódi véletlen számokat adnak 0—255 között.

```
1 POKE 54287,100:
   REM hangmagasság beállítva 100-ra
2 POKE 54290,129:
   REM fehér susogás
3 PRINT PEEK (54299)
4 GOTO 3: REM mégegyszer
```

A fehér susogás a rózsaszín susogással ellentétben az egész frekvenciatartományban véletlenszerű frekvenciaeloszlást vált ki.

Nincs hang és időkijelzés

Ha a C64 nem ad ki hangot, és a két CIA valósidejű órája sem működik, akkor nagy valószínűséggel, szinte biztosan a váltakozó feszültségű áramellátás szakadt meg.

Az okot könnyű megszüntetni, mivel ebben az esetben az okozta a hibát, hogy bekapcsolt számítógépbe dugták be a magnó (Datasette) csatlakozóját. Ez rövidzárlatot okozott, és a számítógépben kiégett a biztosíték. Ennek a kicserélése után minden kifogástalanul fog működni.

Az Országos Commodore Egyesület szolgáltatásai

Egyesületi tagoknak 20% kedvezmény:

VC—20 memóriabővítés 3—27 kByte-os:	kiépítéstől függő
C—16, C—116 memóriájának bővítése 64 kByte-ra:	3500 Ft
C—16 belső 16 kByte-os EPROM bővítés:	1450 Ft
C—16 belső 32 kByte-os EPROM bővítés:	2900 Ft
C—16 belső 8 kByte-os SOFT—ROM bővítés:	2800 Ft
C—16 belső 32 kByte-os SOFT—ROM bővítés:	4000 Ft
C—16 8 kByte-ról 32 kByte-ra átalakítás:	2000 Ft
C—16 és 1541 kompatibilis lemezegység párhuzamosítása:	3200 Ft
SOFTROM modul 32K, kikapcsoláskor sem felejt C-16, C-116, +4	5000 Ft
FÉK C—16, C—116, +4 potméteres sebességváltoztatás 0%-tól 100%-ig fokozatmentesen	2000 Ft
TTL IC-teszter (Cartridge+lemezen a program)	4300 Ft
+4, C—16, C—116 UNI—ROM modul különféle kiépítésekben:	
— 8 kByte SOFT—ROM	3400 Ft
— 16 kByte SOFT—ROM	4000 Ft
— 8 kByte SOFT—ROM 16 kByte EPROM	4400 Ft
— 16 kByte SOFT—ROM 16 kByte EPROM	5000 Ft
— 16 kByte EPROM	2200 Ft

Egyesület tagoknak 30% kedvezmény:

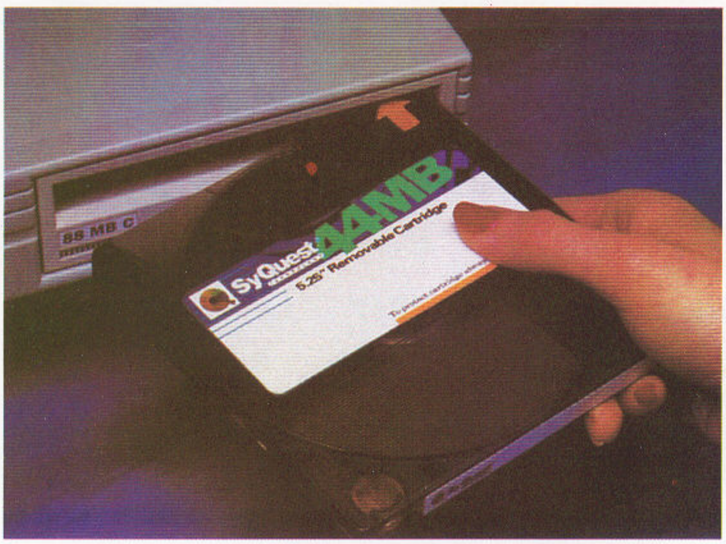
Speeddos (átkapcsolható) operációs rendszer beépítése (C64 átalakítás, lemezegység átalakítás + párhuzamos kábel)	5000 Ft
1541 kompatibilis lemezegységbe elektronikus lemezlyukasztó beépítése	900 Ft
PAGEFOX magyar ékezetes kiadvány-szerkesztő cartridge (a teljes A/4-es oldal kinyomtatásához 640 pont/soros nyomtató szükséges minimum, pl. Citizen 120D)	5500 Ft
FASTLOAD cartridge (lemezes gyorstöltő, másoló, monitor)	1500 Ft
TTL IC-teszter cartridge + program	4300 FT
288/256 Kbyte-os eprombank (vezérlő eprommal)	5000 Ft
Eprométegető (2716-tól 27512-ig)	5000 Ft
C64-hez tároló oszcilloszkóp	8000 Ft
C64-bővítő-port elosztó (egyszerre 4 db cartridge lehet a gépben, melyeket gombnyomásra lehet kapcsolni)	7500 Ft
C64 USER — CENTRONICS nyomtatókábel (GEOS kábel)	1500 Ft
256 K RAM-diszk (256 Kbyte RAM-mal)	14000 Ft
256 K RAM-diszk (64 Kbyte RAM-mal)	9000 Ft
2×64 Kbyte-os cartridge igény szerinti programokkal feltöltve	4.300 Ft
64 Kbyte-os cartridge igény szerinti programokkal feltöltve	3000 Ft
Képújság (teletext) dekóder C-64-re	10000 Ft
1764-es RAM bővítő GEOS-hoz is használható	11250 Ft
Epromok programozása meglévő programokkal, vagy saját hozott programok beégetésével 2716-tól 27512-ig az epprommal együtt egységesen	700 Ft

Áraink az alkatrészárak változásai befolyásolhatják.

A fenti bővítések megrendelhetők levélben az O.C.E. címén, valamint személyesen a havonta rendezendő klubdélelőttön, ahol rendszeres bemutatót is tartunk.

Postázás esetén 100 Ft postaköltség kerül felszámításra.

A kedvezmény igénybevételéhez az O.C.E. tagsági igazolvány bemutatása szükséges.



SyQuest TECHNOLOGY

Hivatalos magyarországi disztribútora:

NOVOTRADE
SZERVIZ Kft.

Cím: 1053 Budapest, Henszlmann I. u. 9
Telefon: 117-4144 Telefax: 117-9692

Cserélhető lemezes winchester!

Szervizeinkben és szerződött viszonteladóinkon keresztül az alábbiakat forgalmazzuk:

SQ 555	44 MB-os $5^{1}/_{4}$	SCSI drive,	SQ 400 lemez	(44 MB)
SQ 5110	88 MB-os $5^{1}/_{4}$	SCSI drive,	SQ 800 lemez	(88 MB)
SQ 3105A	105 MB-os $3^{1}/_{2}$	AT-BUS drive,	SQ 310 lemez	(105 MB)
SQ 2542A	42 MB-os $2^{1}/_{2}$	AT-BUS drive,	SQ 240 lemez	(42 MB)

Számítógépszerviz !

*Személyi számítógépek és perifériák javítását, karbantartását továbbra is tőlünk rendelje meg! Gyors, pontos és megbízható! Örök garanciát kap, ha **átalánydíjas javítási és karbantartási** szerződésformánkkal védi nagyértékű gépei műszaki állapotát.*

Várjuk jelentkezését az alábbi szervizeinkben:

1083	Budapest,	Szigony u. 8.	Tel.: 1343-153
3525	Miskolc,	Fazekas u. 1–3.	Tel.: 46/321-488
5600	Békéscsaba,	Bartók B. u. 37.	Tel.: 66/327-195
6724	Szeged,	Csongrádi sgt. 76.	Tel.: 62/493-185
9700	Szombathely,	Szalonak u. 31.	Tel.: 94/314-519

NOVOTRADE SZERVIZ KFT.

SZERENCSEJÁTÉK RT

Ára: 156 Ft

ACOMP

SZÁMÍTÁSTECHNIKAI KFT.

1141 Budapest, Álmos vezér útja 17., Tel.: 183-1817, Fax: 251-2523

Commodore Amiga 500	28 900 Ft
Commodore Amiga 500 Plus	29 900 Ft
Commodore Amiga 600	24 990 Ft
Commodore Amiga 600/40 MB HD	53 500 Ft
Commodore Amiga 1200	49 990 Ft
Commodore Amiga 1200/40 MB HD	74 900 Ft
Commodore Amiga 1200/80 MB HD	87 900 Ft
Commodore Amiga 4000/040/6MB/120MB	269 000 Ft
+ 4 MB RAM modul	29 000 Ft
Commodore 1942 multisync Monitor	51 900 Ft
Commodore 1084 s Stereo-Color monitor	28 900 Ft
Commodore 1085 s Stereo-Color monitor	27 900 Ft
Commodore A-520 TV-Modulator	3 500 Ft
Commodore C-64 II	8 890 Ft
Commodore C-64 + Joystick + játék	9 390 Ft
Commodore 1541 II Floppy drive	9 990 Ft
Commodore C-128D	24 990 Ft
Commodore 1802 monitor C-64-hez	24 990 Ft
Commodore Datassette	1 750 Ft
Commodore MPS 1230 printer	19 800 Ft
512 Kb órás memóriabővítő	3 200 Ft
2.0 Mb órás memóriabővítő	9 900 Ft
1.0 Mb-os chip bővítő Amiga 500 Plus-ba	5 500 Ft
1.0 Mb-os órás chip bővítő Amiga 600-ba	5 900 Ft
1 MB/4 MB Fast bővítő Amiga 1200-ba	? ??? Ft
Quickshot II Plus Joystick	640 Ft
THE BUG mikrokapcsolós profi Joystick	2990 Ft
Quickshot QS – 123 analóg joystick	940 Ft
Mouse pad	200 Ft
Noris üveg 14" monitorfilter	1 990 Ft
Noris hálós 14" monitorfilter	590 Ft
Noris MB 80 3,5" lemeztartó	390 Ft
Noris DB 100 5,25" lemeztartó	390 Ft
4 Player Adapter(4 Joystick csatlakozó)	1890 Ft

NoName 3.5" DSDD lemez	450 Ft
NoName 3.5" DSHD lemez	680 Ft
NoName 5.25" DSDD lemez	190 Ft
NoName 5.25" DSHD lemez	360 Ft
Maxell 3.5" MF2-DD lemez	890 Ft
Maxell 3.5" MF2-HD lemez	1 550 Ft
Maxell 5.25" MD2-HD lemez	790 Ft
Profex 3.5" DSDD lemez (11 db/Form)	760 Ft
Profex 3.5" DSHD lemez (11 db/Form)	1 090 Ft
Fuji 5.25" MD 2DD lemez	360 Ft
Fuji 5,25" MF 2HD lemez	690 Ft
Action Replay MK III. (Amiga) + könyv	9 990 Ft
Action Replay MK VI. (C-64) kézikönyvvel	5 900 Ft
C-64 midi szoftverrel	6 500 Ft
C-64/C-128 mouse	2 500 Ft
Swifty Amiga/Atari Mouse (3 gombos)	2 500 Ft
Midi Amiga Interface	2 990 Ft
Handyscanner Amigához	13 900 Ft
Boot Selector Amigához	1 490 Ft
Stereo hangdigitalizáló Amigához	6 900 Ft
Trackball Amigához	3 590 Ft
Sound Blaster PRO-2 Deluxe	18 900 Ft
Sound Blaster 16 ASP Deluxe	26 900 Ft
Rocgen Genlock for Amiga	9 900 Ft
Rockey Advanced Video Keying for Amiga	24 900 Ft
Roctec 3,5" Amiga slim külső dirve	9 490 Ft
Roctec 3,5" Ivory Anti-Vírus Amiga drive	9 990 Ft
Roctec 3,5" Black Anti-Vírus Amiga drive	9 990 Ft
Roctec 3,5" Amiga belső drive	8 500 Ft
Rochard HD kontroller A500/A500+	17 900 Ft
+ 80 MB Hard Disk	42 900 Ft
+ 1 Mbyte SIMM Ram	4 500 Ft
Amiga Magazin (német) újság	450 Ft
Power Play (német) újság	450 Ft

Áraink az 1 év garanciát és az ÁFA-t tartalmazzák!

Július elején címünk megváltozik:
1135 Budapest, Szent László u. 74/A. Telefon: 149-6165

Üzletünk megnyitása alkalmából szenzációs árengedményekkel kedveskedünk vásárlóinknak